# MERLIN - 2.0 PLUS

**Φορητό περιβάλλον ελαχιστοποίησης**

**Εγχειρίδιο χρήσης**

# Πρόλογος

Η ενασχόληση των ερευνητών της ομάδος με προβλήματα ελαχιστοποίησης ξεκίνησε στις αρχές της δεκαετίας του 80 με την ανάπτυξη αυτόνομων υποπρογραμμάτων ελαχιστοποίησης. Βρήκαμε ότι η τακτική αυτή δεν ήταν πολύ αποδοτική αφού μιά μέθοδος μπορούσε να αποδόσει καλά σε ένα πρόβλημα και άσχημα σε κάποιο άλλο. Αυτό μας δίδαξε ότι πρέπει να χρησιμοποιούμε παραπάνω από μία μεθόδους ελαχιστοποίησης στα προγράμματά μας, ακόμη και αν το πρόβλημα φαινόταν απλό. Τότε τα ερωτήματα που παρουσιάστηκαν ήταν ποιές μεθόδους να χρησιμοποιήσουμε, πως μπορούσαμε να συγκρίνουμε την απόδοσή τους με ένα εύκολο και ομοιογενή τρόπο κλπ. Η κωδικοποίηση του προγράμματος οδηγού και η κλήση των απαραίτητων υποπρογραμμάτων κάθε φορά που είχαμε ένα διαφορετικό πρόβλημα ήταν μία επίπονη διαδικασία. Επειδή ασχολούμαστε συχνά με προβλήματα που απαιτούν την επίλυση μη γραμμικών συστημάτων (που με τη σειρά τους μπορούν να αναχθούν σε προβλήματα ελαχιστοποίησης) και άλλα προβλήματα που απαιτούν ελαχιστοποίηση, συνειδητοποιήσαμε ότι θα ήταν βολικό να αναπτύξουμε ένα περιβάλλον όπου το μόνο που θα χρειαζόταν να προγραμματιστεί θα ήταν η αντικειμενική συνάρτηση.

Αρχικά αναπτύξαμε ένα συμπαγές πρόγραμμα που ονομαζόταν minapl (αδημοσίευτο), όπου δοκιμάσαμε διάφορους αλγορίθμους και πειραματιστήκαμε με διάφορους τρόπους για να συνδέσουμε τα διάφορα κομμάτια μεταξύ τους. Επειδή πολλές μέθοδοι ελαχιστοποίησης απαιτούν τις πρώτες παραγώγους της συνάρτησης, αναπτύξαμε ένα αξιόπιστο (αλλά δαπανηρό) τρόπο γιά την αριθμητική τους προσέγγιση. Επίσης προσθέσαμε δυνατότητες που νομίσαμε ότι ήταν χρήσιμες. Για παράδειγμα, ο χρήστης μπορεί να θέλει να καθηλώσει την τιμή μιας μεταβλητής σε συγκεκριμένη τιμή, να θέσει όρια στις μεταβλητές, κλπ.

Έχοντας αποκτήσει όλη αυτή την εμπειρία αποφασίσαμε ότι ήμαστε έτοιμοι να αναπτύξουμε ένα νέο πρόγραμμα που θα συνδύαζε ταυτόχρονα ευκολία χρήσης και αποτελεσματικότητα. Η κωδικοποίηση έγινε με "ανοικτό" τρόπο, έτσι ώστε ο χρήστης να μπορεί να το διευρύνει εύκολα προσθέτοντας δικές του εντολές και υποπρογράμματα. Αυτό ήταν το Merlin-1.0 [1].

Μετά καταλάβαμε τη χρησιμότητα για μία γλώσσα προγραμματισμού που θα καθοδηγούσε το Merlin χωρίς την παρεμβολή του χρήστη και θα αυτοματοποιούσε τη διαδικασία ανάπτυξης στρατηγικών για την επίλυση πιό σύνθετων προβλημάτων. Έτσι αναπτύχθηκε η γλώσσα προγραμματισμού MCL (Merlin Control Language) [2] και η προγραμματιζόμενη έκδοση Merlin-2.0 [3]. Τέλος η έκδοση διπλής ακρίβειας Merlin-2.1 [4] παρουσιάστηκε για μηχανές με μήκος λέξης 32 bits.

# Περιεχόμενα


1. Εισαγωγή....6
    1.1 Τι προβλήματα λύνει το Merlin....6
    1.2 Συμβολισμός....8
2. Εγκατάσταση....9
    2.1 Διανομή του λογισμικού....9
    2.2 Εγκατάσταση του Merlin....9
    2.3 Εγκατάσταση της γλώσσας προγραμματισμού MCL....12
3. Τι πρέπει να προγραμματίσει ο χρήστης....13
    3.1 Η αντικειμενική συνάρτηση....13
    3.2 Οι παράγωγοι....14
    3.3 Ειδική μορφή της αντικειμενικής συνάρτησης. Άθροισμα τετραγώνων....15
    3.4 Το κυρίως πρόγραμμα....16
    3.5 Δοκιμάζοντας το πακέτο....17
4. Βασική χρήση του Merlin....19
    4.1 Το λειτουργικό σύστημα του Merlin....19
    4.2 Τρόποι αναφοράς στις παραμέτρους της αντικειμενικής συνάρτησης....19
    4.3 Ανάθεση αρχικών τιμών. Η εντολή POINT....20
    4.4 Οι εντολές SHORTDIS και VALDIS....21
    4.5 Η διαδικασία της ελαχιστοποίησης. Πίνακες ελέγχου....21
    4.6 Τερματισμός του προγράμματος....24
5. Παράγωγοι....25
    5.1 Τρόποι υπολογισμού των παραγώγων....25
    5.2 Έλεγχος των παραγώγων. Εντολές GERROR, GRADDIS και GRADCHECK....25
    5.3 Αυτόματη αλλαγή τρόπου παραγώγισης. Η εντολή MAD....27
6. Εντολές ελέγχου....29
    6.1 Έλεγχος του τρόπου λειτουργίας του Merlin. Εντολές MODEDIS και LIMITS....29
    6.2 Μορφή της αντικειμενικής συνάρτησης. Εντολές SOS και GENERAL....30
    6.3 Τρόπος λειτουργίας. Εντολές IAF και BATCH....30
    6.4 Έλεγχος εξόδου. Εντολές FULLPRINT, HALFPRINT και NOPRINT....31
    6.5 Μετρητές κλήσεων. Εντολή RESET....31
    6.6 Παρουσίαση των πινάκων ελέγχου. Εντολές PANELON και PANELOFF....31
    6.7 Μετονομασίες εντολών. Εντολές ALIAS, UNALIAS και ALIASDIS....32
    6.8 Απαγόρευση υπολογισμού της συνάρτησης. Εντολές NOEVAL και EVALUATE....33
    6.9 Ανάθεση συμβολικών ονομάτων. Οι εντολές GODFATHER, NAME και INDEX....33
    6.10 Καθορισμός του συνόλου χαρακτήρων. Η εντολή CSET....34
    6.11 Ανακατεύθυνση της εξόδου. Εντολές HIDEOUT και REVEAL....34

6.12 Έλεγχος βημάτων. Εντολές STEP, STEPDIS, STEPALL και ADJUST ..... 34
7. Άλλες δυνατότητες ..... 36
7.1 Όρια στις παραμέτρους. Εντολές MARGIN και DEMARGIN ..... 36
7.2 Σταθεροποίηση των παραμέτρων. Εντολές FIX, LOOSE, FIXALL, LOOSALL ..... 37
7.3 Συσσώρευση αρχικών σημείων. Η εντολή ACCUM ..... 39
7.4 Γραφική απεικόνιση. Η εντολή GRAPH ..... 40
7.5 Μαζική ανάθεση αρχικών τιμών. Η εντολή INIT ..... 41
8. Αρχεία ..... 43
8.1 Το αρχείο BACKUP. Εντολές NOBACK, LASTBACK και FULLBACK ..... 43
8.2 Το αρχείο DATA. Εντολές DATASKIP και DATAWRITE ..... 44
8.3 Το αρχείο DISPO. Εντολές DISPOSKIP και DISPOWRITE ..... 44
8.4 Το αρχείο STORE. Η εντολή MEMO ..... 44
8.5 Το αρχείο INIPO ..... 44
8.6 Επιθεώρηση των αρχείων. Οι εντολές INSPECT και PICK ..... 44
8.7 Η εντολή DISCARD ..... 45
8.8 Το αρχείο HELP. Οι εντολές HELP, INTRODUCE και LIST ..... 45
8.9 Τα αρχεία MACROF και CONFIG ..... 46
8.10 Η εντολή CATALOG ..... 46
9. Παραμετρικά εντολοπακέτα ..... 47
9.1 Είδη εντολοπακέτων ..... 47
9.2 Κατασκευή εντολοπακέτων. Η εντολή MACRO ..... 48
9.3 Κλήση των εντολοπακέτων. Οι εντολές DISKON, DISKOFF ..... 49
9.4 Παραμετρικά εντολοπακέτα ..... 49
10. Οι αλγόριθμοι ελαχιστοποίησης ..... 51
10.1 Συμβολισμός ..... 51
10.2 Τερματισμός των ελαχιστοποιητών. Οι εντολές TARGET και NOTARGET ..... 51
10.3 Κοινές παράμετροι ..... 52
10.4 BFGS ..... 53
10.5 DFP ..... 54
10.6 CONGRA ..... 55
10.7 ROLL ..... 56
10.8 RANDOM ..... 57
10.9 SIMPLEX ..... 58
10.10 TOLMIN ..... 59
10.11 LEVE ..... 60
10.12 AUTO ..... 60
11. Η γλώσσα προγραμματισμού MCL ..... 61
11.1 Τι είναι η γλώσσα MCL ..... 61
11.2 Ο μεταγλωτιστής MCL ..... 61
11.3 Εκτέλεση προγραμμάτων MCL. Η εντολή RUNMCL ..... 63

11.4 Οι εντολές LEVE και TOLMIN ..... 63
12. Διαμόρφωση του συστήματος ..... 65
12.1 BIGGER ..... 65
12.2 SMALLER ..... 65
12.3 LIMIT ..... 66
12.4 INPUT_UNIT ..... 66
12.5 OUTPUT_UNIT ..... 66
12.6 INPUT_FILE ..... 67
12.7 OUTPUT_FILE ..... 67
12.8 PRECONNECT ..... 67
12.9 NOPRECONNECT ..... 68
12.10 UNIT_RANGE ..... 68
12.11 NOHEADER ..... 68
12.12 HEADER ..... 68
12.13 CSET ..... 69
12.14 PRINTOUT ..... 69
12.15 PROLOG ..... 69
12.16 EPILOG ..... 70
12.17 MODE ..... 70
12.18 FILE ..... 70
12.19 VAR ..... 71
12.20 TERMS ..... 71
12.21 Σχόλια ..... 71
13. Επεκτάσεις ..... 72
13.1 Μέγιστος αριθμός παραμέτρων και όρων ..... 72
Βιβλιογραφία ..... 73
Παράρτημα Α. Οι εντολές Merlin ..... 75
Παράρτημα Β. Ευρετήριο πινάκων ..... 77

# Κεφάλαιο 1

# Εισαγωγή

## 1.1 Τι προβλήματα λύνει το Merlin

Η ελαχιστοποίηση μιάς πολυδιάστατης συνάρτησης είναι ένα πρόβλημα κοινό σε πολλά ερευνητικά πεδία. Πλήθος προβλημάτων στους τομείς της μηχανολογίας, φυσικής, χημείας κα., ανάγονται συχνά στην επίλυση ενός προβλήματος ελαχιστοποίησης. Ως παραδείγματα αναφέρουμε την επίλυση συστημάτων μη γραμμικών εξισώσεων, παραλακτικές μεθόδους και την προσαρμογή δεδομένων σε αναλυτικές συναρτήσεις, που είναι μιά καθημερινή ανάγκη στην πειραματική έρευνα. Οι δυσκολίες της πολυδιάστατης ελαχιστοποίησης είναι πολλές και ποικίλες (ο αναγνώστης παραπέμπεται για ενημέρωση στις αναφορές [5-10]). Είναι γνωστό ότι δεν υπάρχει μέθοδος που να αντιμετωπίζει ικανοποιητικά όλα τα προβλήματα. Η συνάρτηση μπορεί να μην έχει γνωστή αναλυτική μορφή, οπότε οι παράγωγοί της πρέπει να εκτιμηθούν αριθμητικά, πράγμα που κοστίζει σε χρόνο υπολογισμού και ακρίβεια. Στην περίπτωση που ταυτόχρονα απαιτείται να ικανοποιούνται γνωστές συνθήκες μεταξύ των μεταβλητών το πρόβλημα αποκτά περιορισμούς και βέβαια γίνεται πιό δύσκολο.

Το Merlin είναι ένα ολοκληρομένο περιβάλλον που κατασκευάστηκε για την επίλυση προβλημάτων ελαχιστοποίησης στοχεύοντας ταυτόχρονα να είναι φιλικό στη χρήση, φορητό μεταξύ διαφορετικών υπολογιστικών συστημάτων και εύκολα επεκτάσιμο. Αντιμετωπίζει προβλήματα ελαχιστοποίησης της μορφής:

Εύρεση τοπικού ελαχίστου της συνάρτησης:

$$f(\mathbf{x}) \qquad \mathbf{x} \in \Re^N \qquad \mathbf{x} = [\, x_1, x_2, \ldots x_N \,]^T$$

υπό τους περιορισμούς ορίων:

$$a_i \le x_i \le b_i \qquad i = 1, 2, \ldots N$$

Ειδική μέριμνα έχει ληφθεί για την αντιμετώπιση προβλημάτων όπου η αντικειμενική συνάρτηση είναι της μορφής:

Εύρεση τοπικού ελαχίστου της συνάρτησης:

$$f(\mathbf{x}) = \sum_{i=1}^{M} f_i(\mathbf{x})^2 \qquad \mathbf{x} \in \Re^N \qquad \mathbf{x} = [\, x_1, x_2, \ldots x_N \,]^T$$

υπό τους περιορισμούς ορίων:

$$a_i \le x_i \le b_i \qquad i = 1, 2, \ldots N$$

Ο χρήστης καθοδηγεί το πρόγραμμα εισάγοντας εντολές, ενώ παρέχεται πλήθος βοηθητικών λειτουργιών. Διαθέτει τη δική του γλώσσα προγραμματισμού (MCL) που επιτρέπει την ανάπτυξη αποτελεσματικών στρατηγικών ελαχιστοποίησης γιά την αντιμετώπιση σύνθετων προβλημάτων. Η κωδικοποίηση σε ANSI-Fortran-77 [10] εξασφαλίζει την φορητότητα μεταξύ διαφορετικών υπολογιστικών συστημάτων. Περίληψη του πακέτου δίνεται στον πίνακα 1. Το Merlin υποστηρίζεται πλήρως, ακόμη και μέσω e-mail. Η διεύθυνση είναι:

```
Internet:     Merlin@cc.uoi.gr
Bitnet:       Merlin@grioanun
```

| Πίνακας 1: Περίληψη των Merlin και MCL. | |
|---|---|
| Τίτλος προγράμματος: | Merlin-2.0-Plus |
| Γλώσσα προγραμματισμού: | ANSI-Fortran-77 |
| Αριθμός γραμμών πηγαίου κώδικα: | 14492 |
| Ακρίβεια: | Απλή ή διπλή ανάλογα με τον τρόπο εγκατάστασης. |
| Ελάχιστες απαιτούμενες λέξεις μνήμης: | 137 Kwords για απλή ακρίβεια. |
| Μέγεθος λέξης: | 32 bits |
| Τίτλος προγράμματος: | MCL-1.0-Plus |
| Γλώσσα προγραμματισμού: | ANSI-Fortran-77 |
| Αριθμός γραμμών πηγαίου κώδικα: | 9623 |
| Ακρίβεια: | Απλή |
| Απαιτούμενες λέξεις μνήμης: | 81 Kwords |
| Μέγεθος λέξης: | 32 bits |

## 1.2 Συμβολισμός

Για την περιγραφή του Merlin χρησιμοποιούνται οι ακόλουθες συμβάσεις:

α) Γραφή

Τα μικρά έντονα γράμματα (**x**, **g** κλπ.) δηλώνουν διανύσματα στο χώρο των N διαστάσεων $\Re^N$, ενώ τα έντονα κεφαλαία (**G**, **H** κλπ.) δηλώνουν πίνακες.

Οι εντολές του Merlin, κώδικας Fortran και είσοδος/έξοδος του υπολογιστή δείχνονται με τη χρήση `μη αναλογικής γραφής Courier`.

Έντονα γράμματα **`μη αναλογικής γραφής Courier`** στα παραδείγματα και τους πίνακες, δείχνουν την είσοδο του χρήστη.

β) Μαθηματικός συμβολισμός

| | |
|---|---|
| $f(\mathbf{x})$ | Η αντικειμενική συνάρτηση. Τα στοιχεία $x_i$ αναφέρονται ως μεταβλητές ή παράμετροι της συνάρτησης. |
| $f_i(\mathbf{x})$ | Οι επιμέρους όροι της αντικειμενικής συνάρτησης, όταν αυτή αποτελεί άθροισμα τετραγώνων. |
| **g** | Το διάνυσμα των πρώτων παραγώγων της συνάρτησης με στοιχεία $g_i = \frac{\partial f}{\partial x_i}$ |
| **G** | Ο πίνακας των δεύτερων παραγώγων της συνάρτησης με στοιχεία $G_{ij} = \frac{\partial^2 f}{\partial x_i \, \partial x_i}$ |
| **H** | Ο αντίστροφος του πίνακα των δευτέρων παραγώγων με στοιχεία $H_{ij} = \left[\mathbf{G}^{-1}\right]_{ij}$ |
| $a_i$, $b_i$ | Τα κάτω και άνω όρια των παραμέτρων $x_i$ αντίστοιχα. Αναφέρονται και ως αριστερά και δεξιά όρια. |
| N | Ο αριθμός των παραμέτρων της συνάρτησης $f(\mathbf{x})$. |
| M | Ο αριθμός των όρων της συνάρτησης $f(\mathbf{x})$, όταν αυτή έχει την ειδική μορφή του αθροίσματος τετραγώνων. |

γ) Άλλες τυπογραφικές συμβάσεις

| | |
|---|---|
| `<cr>` | Δηλώνει πάτημα του πλήκτρου Return. |
| `<space>` | Δηλώνει τον κενό χαρακτήρα. |
| `...` | Δηλώνει ότι τα σύμβολα που προηγούνται μπορεί να επαναληφθούν. |
| [1] | Οι αριθμοί σε αγκύλες αποτελούν βιβλιογραφικές αναφορές. |

# Κεφάλαιο 2

# Εγκατάσταση

## 2.1 Διανομή του λογισμικού

Το λογισμικό διατίθεται στα εξής μέσα:

1) Δύο δισκέτες 1.44Mb, MS-DOS format.
2) Δύο δισκέτες 800Kb, Macintosh format.
3) Δύο δισκέτες 1.44Mb, Sun-format.
4) Συμπιεσμένο αρχείο tar (270 Kb) για μεταφορά μέσω ftp, mail, κλπ.

Τα αρχεία που περιέχονται είναι:

| | |
|---|---|
| `merlin.f` | Ο πηγαίος κώδικας του Merlin |
| `mcl.f` | Ο πηγαίος κώδικας του μεταγλωτιστή MCL. |
| `HELP` | Αρχείο άμεσης βοήθειας του Merlin. |
| `DESC` | Αρχείο οδηγιών για τον μεταγλωτιστή MCL. |
| `Fun/rosen.f` | Η συνάρτηση Rosenbrock και οι παράγωγοί της. |
| `Fun/morefun.f` | Οι 35 δοκιμαστικές συναρτήσεις που περιέχονται στο [11]. |
| `Driv/std.f` | Υπόδειγμα του κυρίως προγράμματος |
| `Driv/more.f` | Το κυρίως πρόγραμα για τις συναρτήσεις στο `morefun.f` |
| `manual.ps` | Το παρόν εγχειρίδιο σε μορφή Postscript. |
| `makefile` | Γιά χρήση με το πρόγραμμα make του unix. Κατασκευάζει τα εκτελέσιμα `test-rosen`, `test-more` και `mcl`. |

## 2.2 Εγκατάσταση του Merlin

Το Merlin έχει εγκατασταθεί και λειτουργήσει σε πλήθος διαφορετικών υπολογιστικών συστημάτων, όπως φαίνεται στον πίνακα 2. Η εγκατάσταση του πακέτου δεν απαιτεί καμία τροποποίηση του πηγαίου κώδικα. Τα ονόματα των αρχείων πρέπει να χρησιμοποιούνται όπως ακριβώς φαίνονται εάν το λειτουργικό σύστημα κάτω από το οποίο θα λειτουργήσει το Merlin θεωρεί διαφορετικά ονόματα που εισάγονται με κεφαλαίους ή πεζούς χαρακτήρες. Οι οδηγίες αναφέρονται σε υπολογιστές με λειτουργικό σύστημα unix · η αντιστοιχία με άλλα λειτουργικά συστήματα είναι προφανής. Τα απαραίτητα βήματα είναι:

1) Το Merlin μπορεί να χρησιμοποιηθεί σε απλή ή διπλή ακρίβεια. Για το σκοπό αυτό σε κάθε υποπρόγραμμα υπάρχει μία δήλωση της μορφής:

```
******IMPLICIT DOUBLE PRECISION (A-H,O-Z)
```

Για να εγκαταστήσετε το πακέτο σε διπλή ακρίβεια, χρησιμοποιήστε τον συντάκτη κειμένων του συστήματος σας και επεμβαίνοντας στο αρχείο `merlin.f` μετατρέψετε την παραπάνω δήλωση σε:

```
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
```

Εάν χρησιμοποιήτε τον καθιερωμένο συντάκτη vi του unix, η απαραίτητη εντολή θα ήταν:

```
:%s/\*\*\*\*\*\*/      /
```

Οι περισσότεροι σύγχρονοι μεταγλωτιστές παρέχουν δυνατότητες αυτόματης μετατροπής ενός κώδικα απλής ακρίβειας σε αντικειμενικό κώδικα διπλής ακρίβειας. Στην περίπτωση αυτή η ανωτέρω αλλαγή κρίνεται περιττή.

2) Χρησιμοποιήστε τον μεταγλωτιστή του συστήματός σας για να παράγετε τον αντικειμενικό κώδικα:

```
f77 -O2 -c merlin.f
```

Σημειώστε ότι το αρχείο `merlin.f` δεν περιέχει τα τμήματα του πηγαίου κώδικα που πρέπει να γραφούν από το χρήστη (αντικειμενική συνάρτηση, κυρίως πρόγραμμα κλπ. Βλέπε κεφάλαιο 3).

3) Δημιουργείστε το αρχείο `CONFIG` που περιγράφει παραμέτρους του συστήματός σας καθώς και προτιμήσεις του χρήστη (για λεπτομερή περιγραφή του αρχείου `CONFIG` βλέπε κεφάλαιο 12).

Συμβουλευτείτε το εγχειρίδιο του μεταγλωτιστή σας για τα παρακάτω:

α) Ο μεγαλύτερος πραγματικός αριθμός (κατά απόλυτη τιμή) που μπορεί να χειριστεί το σύστημά σας (`BIGGER`). Η προκαθορισμένη τιμή είναι 1.E+38.

β) Ο μικρότερος πραγματικός αριθμός (κατά απόλυτη τιμή) που μπορεί να χειριστεί το σύστημά σας, χωρίς να προκύψει κατάσταση underflow (`SMALLER`). Η προκαθορισμένη τιμή είναι 1.E-36.

γ) Οι αριθμοί των μονάδων Fortran που χρησιμοποιεί ο μεταγλωτιστής για είσοδο/έξοδο (`INPUT_UNIT` και `OUTPUT_UNIT`). Οι προκαθορισμένες τιμές είναι 5 και 6 αντίστοιχα. Εάν ο μεταγλωτιστής δεν χρησιμοποιεί προκαθορισμένες μονάδες για είσοδο/έξοδο παραλείψτε τις οδηγίες `INPUT_UNIT` και `OUTPUT_UNIT`.

δ) Εάν οι ανωτέρω μονάδες είναι προσυνδεμένες από το λειτουργικό σύστημα με το πληκτρολόγιο και την οθόνη. Δηλαδή εάν Fortan εντολές της μορφής:

```
READ (INPUT_UNIT,*)
WRITE (OUTPUT_UNIT,*)
```

αναφέρονται στο πληκτρολόγιο και στην οθόνη αντίστοιχα. Σε αυτή την περίπτωση χρησιμοποιήστε την οδηγία `PRECONNECT`, την `NOPRECONNECT` ειδάλως. Η προκαθορισμένμη τιμή είναι `PRECONNECT`.

ε) Το σύνολο χαρακτήρων που χρησιμοποιεί το σύστημά σας. Το Merlin υποστηρίζει δύο σύνολα χαρακτήρων: `ASCII` και `EBCDIC`. Αν ο υπολογιστής σας χρησιμοποιεί κάποιο άλλο εισάγετε την οδηγία `CSET NORMAL`. Σε αυτή την περίπτωση όμως πεζοί χαρακτήρες δεν θα αναγνωρίζονται και όλη η είσοδος του Merlin πρέπει να δίνεται με κεφαλαίους χαρακτήρες. Η προκαθορισμένη τιμή είναι `ASCII`.

Επίσης θα πρέπει να αποφασίσετε για τα παρακάτω:

στ) Η προκαθορισμένη τιμή των ορίων των παραμέτρων $x_i$, όταν ο χρήστης δεν έχει εισάγει όρια γιά αυτές (`LIMIT`). Η τιμή αυτή δεν είναι κρίσιμη και πρέπει να είναι μερικές τάξεις μεγέθους μικρότερη από τον αριθμό `BIGGER`. Η προκαθορισμένη τιμή είναι 1.E+30.

ζ) Εάν ο τρόπος χρήσης του πακέτου θα είναι κυρίως διαλογικός. Εδώ δεν υπάρχει προκαθορισμένη τιμή. Εάν δεν χρησιμοποιηθεί η κατάλληλη οδηγία διαμόρφωσης `MODE`, ο χρήστης θα ερωτηθεί διαλογικά κατά την εκκίνηση του Merlin, για τον επιθυμητό τρόπο λειτουργίας.

Εισάγετε όλες τις παραπάνω πληροφορίες στο αρχείο `CONFIG`. Τιμές που συμπίπτουν με τις προκαθορισμένες τιμές (που είναι κατάλληλες για τους περισσότερους μεταγλωτιστές σε απλή ακρίβεια) μπορούν να παραλειφθούν. Το αρχείο `CONFIG` πρέπει να έχει τη μορφή:

```
BIGGER 1.D300
SMALLER 1.D-200
INPUT_UNIT 5
OUTPUT_UNIT 6
PRECONNECT
CSET ASCII
LIMIT 1.D280
MODE IAF
```

4) Στο σημείο αυτό η εγκατάσταση του Merlin έχει ολοκληρωθεί. Τα αρχεία που θα πρέπει να έχετε διαθέσιμα μέχρι στιγμής είναι:

α) `merlin.o` Αντικειμενικός κώδικας του Merlin. Δεν περιέχονται τα υποπρογράμματα funmin, granal, subsum καθώς και το κυρίως πρόγραμμα. Όλα αυτά απαιτείται να γραφούν από το χρήστη. Η μορφή τους εξηγείται στο κεφάλαιο 3.

β) `CONFIG` Αρχείο διαμόρφωσης για το σύστημά σας.

γ) `HELP` Αρχείο βοήθειας.

Σημειώστε ότι εάν το αρχείο `CONFIG` δεν υπάρχει χρησιμοποιούνται οι προκαθορισμένες τιμές του προγράμματος (βλέπε κεφάλαιο 12), ενώ εάν το αρχείο `HELP` δεν υπάρχει, δεν είναι δυνατή η παροχή διαλογικής βοήθειας κατά τη διάρκεια της χρήσης του Merlin.

## 2.3 Εγκατάσταση της γλώσσας προγραμματισμού MCL

Τα απαραίτητα βήματα είναι:

1) Στο αρχείο `mcl.f` περιέχεται ο μεταγλωτιστής της MCL. Το μόνο που απαιτείται είναι η παραγωγή του αντίστοιχου εκτελέσιμου προγράμματος:

```
f77 -O2 -o mcl mcl.f
```

2) Στο σημείο αυτό η εγκατάσταση της MCL έχει ολοκληρωθεί. Τα αρχεία που θα πρέπει να έχετε διαθέσιμα είναι:

   α) `mcl` Το εκτελέσιμο πρόγραμμα.

   β) `DESC` Αρχείο με οδηγίες για τον μεταγλωτιστή MCL.

   Σημειώστε ότι εάν το αρχείο `DESC` δεν υπάρχει, πολλές από τις εντολές του Merlin δεν θα αναγνωρίζονται από την MCL και κατά συνέπεια το πακέτο θα υπολειτουργεί.

Πίνακας 2: Υπολογιστικά συστήματα στα οποία εγκαταστάθηκε και λειτούργησε το πακέτο Merlin/MCL σε απλή και διπλή ακρίβεια.

| Υπολογιστής | Λειτουργικό Σύστημα | Μεταγλωτιστής Fortran |
|---|---|---|
| Sun Sparc-classic | SunOS 4.1.3 | f77 v. 1.4 |
| Sun Sparc-classic | SunOS 4.1.3 | f2c v. 19940714 [12] |
| Control Data cd4680 | EP/IX 1.4.3 MP | f77 v. 2.20 |
| Control Data cd4680 | EP/IX 1.4.3 MP | pftn v. 1.2 |
| Silicon Graphics Indy | IRIX 5.1.1.1 | f77 |
| Macintosh | System 7.0.1.1 | Language Systems Fortran 3.3 |
| Macintosh | System 7.0.1.1 | Absoft Mac Fortran 2.4 |
| T800 Transputer | ---------- | 3L Parallel Fortran |

# Κεφάλαιο 3

# Τι πρέπει να προγραμματίσει ο χρήστης

Το Merlin έχει σχεδιαστεί έτσι ώστε να είναι φορητό και εύκολα προσαρμόσιμο σε ένα πλήθος δυνατών εφαρμογών. Πρίν τη χρήση του πακέτου ο χρήστης ανάλογα με τις ανάγκες του, πρέπει να προετοιμάσει (σε μορφή πηγαίου κώδικα) μιά σειρά από υποπρογράμματα.

## 3.1 Η αντικειμενική συνάρτηση

Πρόκειται για τη συνάρτηση της οποίας αναζητούμε ελάχιστο. Πρέπει να δηλωθεί ώς:

```
FUNCTION FUNMIN ( X, N )
DIMENSION X(N)
```

Τα ορίσματα της συνάρτησης είναι:

| | | |
|---|---|---|
| `X` | - είσοδος - | Μονοδιάστατος πίνακας με τις τιμές των παραμέτρων $x_i$. |
| `N` | - είσοδος - | Ο αριθμός των παραμέτρων της συνάρτησης. |
| `FUNMIN` | - έξοδος - | Η τιμή της αντικειμενικής συνάρτησης f(**x**). |

Υπόδειγμα της παραπάνω συνάρτησης funmin γιά τη συνάρτηση Rosenbrock [13]:

$$f(x_1, x_2) = 100\,(x_2 - x_1^2)^2 + (1 - x_1)^2$$

φαίνεται στον πίνακα 3.

Πίνακας 3: Υπόδειγμα της συνάρτησης funmin. Έχει κωδικοποιηθεί η συνάρτηση Rosenbrock.

```
FUNCTION FUNMIN ( X, N )
DIMENSION X(N)
X1 = X(1)
X2 = X(2)
FUNMIN = 100.*(X2-X1**2)**2 + (1.-X1)**2
END
```

## 3.2 Οι παράγωγοι

Εάν είναι αναλυτικά γνωστές οι πρώτες παράγωγοι της συνάρτησης (όλες, ή ορισμένες μόνο), αυτές μπορεί να προγραμματιστούν:

```
SUBROUTINE GRANAL ( N, X, VAL, GRAD )
DIMENSION X(N), GRAD(N)
```

Τα ορίσματα του υποπρογράμματος είναι:

| | | |
|---|---|---|
| N | - είσοδος - | Ο αριθμός των παραμέτρων της συνάρτησης. |
| X | - είσοδος - | Μονοδιάστατος πίνακας με τις τιμές των παραμέτρων $x_i$. |
| VAL | - είσοδος - | Η τιμή της συνάρτησης στο σημείο **x**. |
| GRAD | - έξοδος - | Μονοδιάστατος πίνακας με τις τιμές των πρώτων παραγώγων $g_i$. |

Υπόδειγμα του υποπρογράμματος granal γιά τη συνάρτηση Rosenbrock δίνεται στον πίνακα 4. Σημειώστε ότι πολλές φορές δεν είναι δυνατόν να έχουμε αναλυτική μορφή παρά μόνο γιά μερικές μόνο από τις παραγώγους της συνάρτησης. Σε αυτή την περίπτωση θα πρέπει να προγραμματίσουμε όσες γνωρίζουμε, ενώ οι υπόλοιπες θα παραλειφθούν. Η ύπαρξη αναλυτικής μορφής γιά μερικές από τις παραγώγους δηλώνεται κατά τη διάρκεια της χρήσης του Merlin με την εντολή `MIXED` (βλέπε παράγραφο 5.1).

Εάν καμία από τις παραγώγους της συνάρτησης δεν είναι αναλυτικά γνωστή, το υποπρόγραμμα granal πρέπει να έχει την τετριμμένη μορφή:

```
SUBROUTINE GRANAL ( N, X, VAL, GRAD )
DIMENSION X(N), GRAD(N)
END
```

Πίνακας 4: Υπόδειγμα του υποπρογράμματος granal. Έχουν κωδικοποιηθεί οι παράγωγοι της συνάρτησης Rosenbrock.

```
SUBROUTINE GRANAL ( N, X, VAL, GRAD )
DIMENSION X(N), GRAD(N)
X1 = X(1)
X2 = X(2)
A = X2-X1*X1
GRAD(1) = -400.*A*X1 - 2.*(1.-X1)
GRAD(2) = 200.*A
END
```

## 3.3 Ειδική μορφή της αντικειμενικής συνάρτησης Άθροισμα τετραγώνων

Εάν η αντικειμενική συνάρτηση είναι της μορφής:

$$f(\mathbf{x}) = \sum_{i=1}^{M} f_i(\mathbf{x})^2$$

μπορεί να προγραμματιστεί και ως υποπρόγραμμα που θα επιστρέφει τις επιμέρους τιμές $f_i$ :

```
SUBROUTINE SUBSUM ( M, N, X, F )
DIMENSION X(N), F(M)
```

Τα ορίσματα του υποπρογράμματος είναι:

| | | |
|---|---|---|
| M | - είσοδος - | Ο αριθμός των επιμέρους όρων της συνάρτησης. |
| N | - είσοδος - | Ο αριθμός των παραμέτρων της συνάρτησης. |
| X | - είσοδος - | Μονοδιάστατος πίνακας με τις τιμές των παραμέτρων $x_i$. |
| F | - έξοδος - | Μονοδιάστατος πίνακας με τις τιμές των επιμέρους όρων $f_i$. |

Υπόδειγμα του υποπρογράμματος subsum γιά τη συνάρτηση Rosenbrock δίνεται στον πίνακα 5. Μόνο μία από τις `FUNCTION FUNMIN` ή `SUBROUTINE SUBSUM` πρέπει να προγραμματιστεί. Σε περίπτωση που η αντικειμενική συνάρτηση είναι άθροισμα τετραγώνων και προγραμματίσουμε την `SUBROUTINE SUBSUM` θα πρέπει αυτό να δηλωθεί κατά τη χρήση του Merlin μέσω της εντολής `SOS` (βλέπε παράγραφο 6.2), ενώ η `FUNCTION FUNMIN` θα πρέπει να έχει τη μορφή:

```
FUNCTION FUNMIN ( X, N )
DIMENSION X(N)
END
```

Σημειώστε ότι αν προγραμματιστεί η `SUBROUTINE SUBSUM`, όλοι οι ελαχιστοποιητές του Merlin μπορούν να τη χρησιμοποιήσου. Η `FUNCTION FUNMIN` δεν μπορεί να χρησιμοποιηθεί από τον ελαχιστοποιητή `LEVE` (βλέπε παράγραφο 10.11), ειδικό για συναρτήσεις που είναι αθροίσματα τετραγώνων.

Εάν η συνάρτηση δεν είναι άθροισμα τετραγώνων και προγραμματίσουμε τη συνάρτηση funmin, τότε το υποπρόγραμμα subsum πρέπει να έχει τη μορφή:

```
SUBROUTINE SUBSUM ( M, N, X, F )
DIMENSION X(N), F(M)
END
```

Πίνακας 5: Υπόδειγμα του υποπρογράμματος subsum. Έχουν κωδικοποιηθεί οι επιμέρους όροι $f_i$ της συνάρτησης Rosenbrock.

```
      SUBROUTINE SUBSUM ( M, N, X, F )
      DIMENSION X(N), F(M)
      X1 = X(1)
      X2 = X(2)
      F(1) = 10.*(X2-X1**2)
      F(2) = 1.-X1
      END
```

## 3.4 Το κυρίως πρόγραμμα

Το Merlin έχει κατασκευαστεί ώς υποπρόγραμμα, έτσι ώστε να είναι δυνατή η κλήση του για την επίλυση προβλημάτων ελαχιστοποίησης μέσα από άλλα προγράμματα. Είναι δηλωμένο ώς:

```
SUBROUTINE MERLIN ( N, M, VERSIM, MAXW, IQUIT )
DIMENSION VERSIM(MAXW)
```

Τα ορίσματα του υποπρογράμματος είναι:

| | | |
|---|---|---|
| N | - είσοδος - | Ο αριθμός των παραμέτρων της συνάρτησης. |
| M | - είσοδος - | Ο αριθμός των επιμέρους όρων της συνάρτησης, όταν αυτή αποτελεί άθροισμα τετραγώνων (διαφορετικά πρέπει να θέσουμε M=0). |
| VERSIM | - είσοδος - | Μονοδιάστατος πίνακας μήκους MAXW που χρησιμοποιείται ώς βοηθητικός χώρος αποθήκευσης από τις μεθόδους BFGS, DFP, TOLMIN, LEVE και SIMPLEX. Το ποσό μνήμης που απαιτείται γιά κάθε μία από αυτές σαν συνάρτηση των N και M φαίνεται στον πίνακα 6. |
| MAXW | - είσοδος - | Το μέγεθος του βοηθητικού πίνακα VERSIM. |
| IQUIT | - έξοδος - | Ακέραια μεταβλητή. Η τιμή της καθορίζεται από το χρήστη όταν δοθεί η εντολή QUIT. |

Υπόδειγμα κλήσης του υποπρογράμματος MERLIN δίνεται στον πίνακα 7.

Πίνακας 6: Απαιτούμενος χώρος μνήμης για τις διάφορες μεθόδους ελαχιστοποίησης (N = αριθμός παραμέτρων, M = αριθμός όρων στο άθροισμα τετραγώνων )

| Μέθοδος | Απαιτούμενες λέξεις μνήμης |
|---|---|
| BFGS | $N^2 + N$ |
| DFP | $N^2 + N$ |
| SIMPLEX | $N^2 + N$ |
| LEVE | $N\ M$ |
| TOLMIN | $N^2 + 11\ N$ |

Πίνακας 7: Υπόδειγμα κυρίως προγράμματος.

```
      PROGRAM MASTER
C  MXV = MAXIMUM NUMBER OF PARAMETERS FOR THE OBJECTIVE FUNCTION.
      PARAMETER ( MXV = 200 )
      PARAMETER ( MAXW = MXV**2+11*MXV )
      DIMENSION VERSIM(MAXW)

      WRITE (*,*) 'ENTER # 0F VARIABLES'
      READ (*,*) N
      WRITE (*,*) 'ENTER # 0F TERMS'
      READ (*,*) M

      CALL MERLIN(N,M,VERSIM,MAXW,IQUIT)
      END
```

## 3.5 Δοκιμάζοντας το πακέτο

Υποδείγματα των funmin, granal και subsum γιά την συνάρτηση Rosenbrock υπάρχουν στο αρχείο `rosen.f`, ενώ υπόδειγμα κυρίως προγράμματος περιέχεται στο αρχείο `std.f`. Σημειώστε ότι αν επιλέξατε να εγκαταστήσετε την έκδοση διπλής ακρίβειας του Merlin όπως περιγράφεται στην παράγραφο 2.2, θα πρέπει επίσης να μετατρέψετε τα αρχεία `rosen.f` και `std.f` σε διπλή ακρίβεια. Η διαδικασία κατασκευής του εκτελέσιμου κώδικα έχει ώς εξής:

1) Μεταγλώτιση των funmin, granal και subsum:

```
f77 -O2 -c rosen.f
```

2) Μεταγλώτιση του κυρίως προγράμματος:

```
f77 -O2 -c std.f
```

3) Σύνδεση όλων των ανωτέρω με τον αντικειμενικό κώδικα του Merlin:

```
f77 -o merlin merlin.o rosen.o std.o
```

Στο σημείο αυτό ο αντικειμενικός κώδικας είναι έτοιμος γιά χρήση. Κατά την εκκίνηση ζητείται από το πρόγραμμα ο αριθμός των παραμέτρων της συνάρτησης (N) και ο αριθμός των όρων (M), όταν αυτή αποτελεί άθροισμα τετραγώνων. Γιά τη συνάρτηση Rosenbrock όπως υπάρχει στο αρχείο `rosen.f` πρέπει να εισάγουμε N=2 και M=2. Επιπλέον, εάν στο αρχείο διαμόρφωσης `CONFIG` δεν έχει καθοριστεί ο τρόπος λειτουργίας με την οδηγία `MODE`, ο χρήστης θα ερωτηθεί διαλογικά για τον τρόπο λειτουργίας. Τα μηνύματα που εμφανίζονται κατά την εκκίνηση του προγράμματος φαίνονται στον πίνακα 8. Ιδιαίτερα σημαντικές είναι οι γραμμές που αναφέρονται στον αριθμούς N και M:

```
 NUMBER OF VARIABLES:     2
 NUMBER OF TERMS:         2
```

και στην ακρίβεια του υπολογιστή, όπως εκτιμήθηκε από το Merlin:

```
 ESTIMATED MACHINE'S ACCURACY: 1.00E-15
```

Στο συγκεκριμένο παράδειγμα χρησιμοποιήθηκε η έκδοση διπλής ακρίβειας. Το προτρεπτικό σήμα:

```
      //////      MERLIN   IS AT YOUR COMMAND !!!
```

δηλώνει ότι το Merlin πλέον αναμένει την εισαγωγή εντολών από το χρήστη.

Πίνακας 8: Τα μηνύματα που παρουσιάζονται κατά την εκκίνηση του Merlin.

```
 ------------------------------------------------------------------------------

  ..........................              M E R L I N - 2.0  PLUS
  ..........................
  ..........................
  ..........................          E-MAIL:  MERLIN @ GRIOANUN.BITNET
  ..........................                   MERLIN @ CC.UOI.GR

 ------------------------------------------------------------------------------
  TO OBTAIN ON-LINE INFORMATION, MAKE SURE THAT
                      FILE:  HELP  , IS PRESENT
 ------------------------------------------------------------------------------
 NUMBER OF VARIABLES:    2
 NUMBER OF TERMS:        2
 ESTIMATED MACHINE'S ACCURACY: 1.00E-15


  ....   W A R N I N G   ....
...  INITIALIZE VARIABLES  ...


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

# Κεφάλαιο 4

# Βασική χρήση του Merlin

## 4.1 Το λειτουργικό σύστημα του Merlin

Όπως έχει ήδη αναφερθεί το Merlin αποτελεί ένα διαλογικό πρόγραμμα που κατευθύνεται από το χρήστη μέσω εντολών. Το λειτουργικό σύστημα του Merlin, δέχεται τις εντολές που εισάγει ο χρήστης, τις αναγνωρίζει, και ενεργεί ανάλογα. Αναγνωρίζονται τα εξής είδη εντολών:

α) Απλές εντολές. Πρόκειται για το βασικό σύνολο εντολών του Merlin που περιγράφονται στα επόμενα κεφάλαια. Μπορούν να εισαχθούν και συντομευμένες στον ελάχιστο αριθμό χαρακτήρων που τις κάνει να ξεχωρίζουν από τις υπόλοιπες. Πχ. η εντολή `SHORTDIS`, μπορεί να εισαχθεί ως `SHORT`, `SHO` ή `SH`, όχι όμως ως `S`, αφού θα προκύψει σύγχιση με άλλες εντολές όπως `STEP`, `SIMPLEX`, κα.
β) Μετονομασίες εντολών. Εξηγούνται στην παράγραφο 6.7.
γ) Κλήσεις παραμετρικών εντολοπακέτων. Εξηγούνται στο κεφάλαιο 9.

## 4.2 Τρόποι αναφοράς στις παραμέτρους της αντικειμενικής συνάρτησης

Κατά τη διάρκεια χρήσης του Merlin, υπάρχει ένας ομοιόμορφος τρόπος αναφοράς στις παραμέτρους της αντικειμενικής συνάρτησης. Όπου λοιπόν χρειάζεται αυτές αναφέρονται με δύο τρόπους:

α) Με τη χρήση ενός ακέραιου δείκτη που παίρνει τιμές από 1 έως N. Ο δείκτης 1 αναφέρεται στην πρώτη παράμετρο $x_1$, ο δείκτης 2 στην δεύτερη $x_2$ , κ.ο.κ. Αυτός ο τρόπος αναφοράς στις παραμέτρους ονομάζεται `INDEX`.
β) Με τη χρήση ενός συμβολικού ονόματος. Κάθε μία από τις παραμέτρους της συνάρτησης μπορεί να αποκτήσει ένα συμβολικό όνομα με τη χρήση της εντολής `GODFATHER` (βλέπε παράγραφο 6.9). Πχ. η πρώτη παράμετρος μπορεί να ονομαστεί `RHO` και η δεύτερη `SIGMA`. Τα συμβολικά ονόματα μπορεί να έχουν μέγεθος μέχρι 10 χαρακτήρες και η επιλογή τους αφήνεται στη απόλυτη διάθεση του χρήστη. Αυτός ο τρόπος αναφοράς στις παραμέτρους ονομάζεται `NAME`.

Μόνο ένας από τους τρόπους αναφοράς `INDEX` και `NAME` μπορεί να χρησιμοποιείται κάθε φορά. Η επιλογή του τρόπου αναφοράς γίνεται μέσω των εντολών `INDEX` και `NAME` με προφανή αντιστοιχία (βλέπε παράγαρφο 6.9). Ο προκαθορισμένος τρόπος λειτουργίας είναι `INDEX`.

## 4.3 Ανάθεση αρχικών τιμών. Η εντολή POINT

Πρίν από κάθε διαδικασία ελαχιστοποίησης πρέπει να δοθούν αρχικές τιμές στις παραμέτρους της συνάρτησης.Ένας τρόπος είναι με τη χρήση της εντολής POINT (υπάρχουν και άλλοι τρόποι, βλέπε τις εντολές INIT, INSPECT και PICK). Η εντολή POINT αναμένει για είσοδο μία γραμμή της μορφής:

```
δείκτης τιμή δείκτης τιμή ...
```

ή στην περίπτωση του τρόπου λειτουργίας NAME:

```
όνομα τιμή όνομα τιμή ...
```

Πχ:

```
1 1.4 2 −2.5
```

ή

```
RHO 1.4 SIGMA −2.5
```

Και στα δύο ανωτέρω παραδείγματα η πρώτη παράμετρος της συνάρτησης λαμβάνει την αρχική τιμή 1.4 ενώ η δεύτερη -2.5, δηλαδή $x_1 = 1.4$ και $x_2 = -2.5$. Ο τρόπος αυτός εισαγωγής δεδομένων ονομάζεται "είσοδος μιάς γραμμής" και συναντάται σε όλες τις εντολές που έχουν σχέση με τις παραμέτρους της συνάρτησης. Εάν δεν επιθυμούμε να εισάγουμε τίποτε, αρκεί ένα απλό πάτημα του <cr>. Αμέσως μετά την "είσοδο μιάς γραμμής" το Merlin ζητά και πάλι τιμές για τις παραμέτρους. Ο πλήρης διάλογος φαίνεται στον πίνακα 9.

Όταν όλες οι παράμετροι της συνάρτησης έχουν πάρει αρχικές τιμές, το Merlin καλεί την αντικειμενική συνάρτηση funmin (ή το υποπρόγραμμα subsum) και υπολογίζει την τιμή της συνάρτησης για τις δεδομένες τιμές των παραμέτρων.

Πίνακας 9: Πλήρης διάλογος ανάθεσης αρχικών τιμών με την εντολή POINT.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
point  <cr>
 POINT
1  1.4  2  -2.5  <cr>
 POINT
<cr>

     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 4.4 Οι εντολές SHORTDIS και VALDIS

Αμέσως μετά την απόδωση αρχικών τιμών ο χρήστης μπορεί να ελέγξει τις τιμές των παραμέτρων και της συνάρτησης με την εντολή SHORTDIS. Τυπικό αποτέλεσμα από τη χρήση της εντολής φαίνεται στον πίνακα 10. Εκτός από τις τιμές των παραμέτρων και της συνάρτησης, εμφανίζεται ο αριθμός κλήσεων της συνάρτησης και ο αριθμός κλήσεων του υποπρογράματος των παραγώγων granal (βλέπε παράγραφο 3.2). Μόνο οι τιμές της συνάρτησης και των μετρητών εμφανίζονται με τη χρήση της εντολής VALDIS.

Πίνακας 10: Τυπικό αποτέλεσμα από τη χρήση της εντολής SHORTDIS.

```
    //////     MERLIN   IS AT YOUR COMMAND !!!
shortdis  <cr>
 SHORTDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                          1               0
 SINCE LAST RESET:               1               0
 ------------------------------------------------------------------------------
   1) RHO       .......   1.4000000000000
   2) SIGMA     .......  -2.5000000000000
         VALUE.......     1989.3200000000

    //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 4.5 Η διαδικασία της ελαχιστοποίησης Πίνακες ελέγχου

Το Merlin διαθέτει ενσωματωμένους 8 αλγορίθμους ελαχιστοποίησης. Καθένας καλείται με τη χρήση αντίστοιχης εντολής, όπως φαίνεται στον πίνακα 11. Γιά κάθε αλγόριθμο, υπάρχει μιά σειρά παραμέτρων που καθορίζει τη λειτουργία του και μπορεί να αλλαχθούν από το χρήστη. Αυτοί ονομάζονται "παράμετροι ελέγχου" της μεθόδου και για την αλλαγή τους το Merlin εμφανίζει ένα "πίνακα ελέγχου". Στον πίνακα ελέγχου παρουσιάζονται αρχικά οι προκαθορισμένες τιμές των παραμέτρων ελέγχου της μεθόδου. Τυπικό παράδειγμα από τη χρήση της BFGS φαίνεται στον πίνακα 12. Ο πίνακας ελέγχου είναι διαφορετικός γιά κάθε μέθοδο και η σημασία των υπό-αλλαγή παραμέτρων ελέγχου για τους ελαχιστοποιητές εξηγείται στο κεφάλαιο 10. Εάν ο χρήστης επιθυμεί να αλλάξει τις τιμές ορισμένων από αυτές θα πρέπει να εισάγει τις αλλαγές του αμέσως μετά την προτροπή:

```
ENTER CHANGES
```

Στο παράδειγμα του πίνακα 12, ο χρήστης άλλαξε την πρώτη παράμετρο (NUMBER OF CALLS: προσεγγιστικός μέγιστος επιτρεπτός αριθμός κλήσεων της αντικειμενικής συνάρτησης) και την τρίτη (ERROR-BOUND: επιτρεπτό σφάλμα κατά τον αριθμητικό υπολογισμό των παραγώγων). Γενικά, αλλαγές στούς πίνακες ελέγχου γίνονται με την εισαγωγή μιάς γραμμής, με τη μορφή:

```
δείκτης_παραμέτρου  νέα_τιμή  δείκτης_παραμέτρου  νέα_τιμή  ...
```

Φυσικά απλό πάτημα του <cr> δηλώνει ότι ο χρήστης δεν επιθυμεί αλλαγές. Οι αλλαγές που γίνονται στους πίνακες ελέγχου διατηρούνται, και οι νέες τιμές που θέτει ο χρήστης εμφανίζονται στις διαδοχικές κλήσεις της ίδιας μεθόδου. Ο πίνακας 13 δείχνει τον πίνακα ελέγχου της BFGS, όπως θα παρουσιαζόταν κατά την δεύτερη κλήση του ελαχιστοποιητή.

Αμέσως μετά την εμφάνιση του πίνακα ελέγχου και τις (πιθανές) αλλαγές από το χρήστη, η επιλεγμένη μέθοδος ελαχιστοποίησης αρχίζει να εκτελείται. Ο χρήστης πληροφορείται την πορεία της ελαχιστοποίησης από μηνύματα της μορφής:

```
LOWER VALUE :     3.88803017613088E+02        33 OUT OF 1000
```

όπου αναφέρεται η τιμή και οι κλήσεις της συνάρτησης που έχουν δαπανηθεί, καθώς και ο προσεγγιστικός μέγιστος αριθμός κλήσεων που ζητήθηκε από το χρήστη. Μετά το πέρας της ελαχιστοποίησης, αναφέρεται ο λόγος τερματισμού της μεθόδου (οι λόγοι τερματισμού περιγράφονται στην παράγραφο 10.2), με μια γραμμή της μορφής:

```
...  CONVERGENCE SLOWER THAN SPECIFIED  ...
```

καθώς και ο αριθμός κλήσεων της αντικειμενικής συνάρτησης που δαπανήθηκαν από τη μέθοδο:

```
NUMBER OF      BFGS  CALLS:        641
```

Η τιμές των παραμέτρων της συνάρτησης μετά το πέρας της ελαχιστοποίησης μπορούν να ελεγθούν με την εντολή SHORTDIS.

Πίνακας 11: Οι αλγόριθμοι ελαχιστοποίησης του Merlin.

| Εντολή | Χρήση πρώτων παραγώγων | Ειδικός γιά ελάχιστα τετράγωνα | Αναφορές |
|---|---|---|---|
| BFGS | √ | | [ 14, 6 ] |
| DFP | √ | | [ 15, 6 ] |
| CONGRA | √ | | [ 8, 16, 6 ] |
| SIMPLEX | | | [ 17, 18 ] |
| ROLL | | | [ 1 ] |
| RANDOM | | | [ 1 ] |
| TOLMIN | √ | | [ 19, 20 ] |
| LEVE | | √ | [ 21, 22, 6 ] |

Πίνακας 12: Τυπικό παράδειγμα χρήσης της μεθόδου BFGS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
bfgs
 BFGS
 ------------------------------------------------------------------------------
                           BFGS - PANEL
 ------------------------------------------------------------------------------
  INDEX    DESCRIPTION                 VALUE       MENU
    1)   NUMBER OF CALLS                300        ANY INTEGER
    2)   TOLERANCE                    1.000E-03    ANY REAL IN (0,1)
    3)   ERROR-BOUND                  1.00E-02     ANY REAL < 1
    4)   USE/RECALCULATE GRADIENT        0         (1/0)
    5)   USE/RESET  HESSIAN             0         (1/0)
    6)   PRINTOUT SELECTION             1         0/1/2
    7)   WALL-PARAMETER                 3         ANY INTEGER
    8)   CANCEL-BUTTON                  1         ( 0/1/2 )
    9)   GRADIENT MODE (FIXED/MAD)      0         (0/1)
   10)   MIN. RELATIVE DROP             0.        ANY REAL IN (0,1)
 ------------------------------------------------------------------------------
    ENTER CHANGES
1  1000  2  5.e-3  <cr>
       LOWER VALUE :    5.40419090538787E-03        22 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    4.59230528016647E-03        37 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    1.09358701726752E-03        53 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    2.07490659926518E-04        64 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    6.32508832515763E-06        80 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    8.36110005888802E-08        94 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    1.57860298524532E-11       107 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    7.56276839203968E-17       119 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    2.76310858005303E-26       132 OUT OF 1000
 ------------------------------------------------------------------------------
       LOWER VALUE :    0.00000000000000E+00       137 OUT OF 1000
 ------------------------------------------------------------------------------

                  ...  CONVERGENCE SLOWER THAN SPECIFIED  ...

 ------------------------------------------------------------------------------
        NUMBER OF      BFGS  CALLS:        137


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

Πίνακας 13: Ο πίνακας ελέγχου της BFGS, όπως θα παρουσιαζόταν κατά την δεύτερη κλήση της μεθόδου.

```
------------------------------------------------------------------------------
                          BFGS - PANEL
 -----------------------------------------------------------------------------
  INDEX    DESCRIPTION                  VALUE       MENU
    1)   NUMBER OF CALLS                1000        ANY INTEGER
    2)   TOLERANCE                     1.000E-03    ANY REAL IN (0,1)
    3)   ERROR-BOUND                   5.00E-03     ANY REAL < 1
    4)   USE/RECALCULATE GRADIENT         0          (1/0)
    5)   USE/RESET  HESSIAN              0          (1/0)
    6)   PRINTOUT SELECTION              1          0/1/2
    7)   WALL-PARAMETER                  3          ANY INTEGER
    8)   CANCEL-BUTTON                   1          ( 0/1/2 )
    9)   GRADIENT MODE (FIXED/MAD)       0          (0/1)
   10)   MIN. RELATIVE DROP               0.         ANY REAL IN (0,1)
 -----------------------------------------------------------------------------
```

## 4.6 Τερματισμός του προγράμματος

Υπάρχουν τρείς διαφορετικές εντολές που τερματίζουν τη λειτουργία του Merlin. Πρίν το Merlin τερματίσει τη λειτουργία του, όλα τα αρχεία που χρησιμοποίησε (βλέπε κεφάλαιο 8) κλείνονται με τη χρήση κατάλληλων Fortran εντολών `close`.

`STOP` Η λειτουργία του Merlin και του καλούντος προγράμματος σταματά αμέσως. (Γίνεται χρήση της εντολής Fortran `STOP`.)

`RETURN` Το Merlin επιστρέφει αμέσως τον έλεγχο στο καλών πρόγραμμα. (Γίνεται χρήση της εντολής Fortran `RETURN`.)

`QUIT` Το Merlin επιστρέφει τον έλεγχο στο καλών πρόγραμμα, όπως και με την εντολή `RETURN`. Προηγουμένως όμως ο χρήστης ερωτάται γιά την τιμή της "μεταβλητής επιστροφής" (ονομάζεται `IQUIT`, βλέπε παράγραφο 3.4). Το καλών πρόγραμμα μπορεί στη συνέχεια να ελέγξει την τιμή της και να καθορίσει ανάλογα τη δράση του.

Εάν χρησιμοποιηθούν οι εντολές `STOP` ή `RETURN`, η τιμή της επιστρεφόμενης μεταβλητής `IQUIT` είναι 0.

# Κεφάλαιο 5

# Παράγωγοι

## 5.1 Τρόποι υπολογισμού των παραγώγων

Στην τρέχουσα έκδοση του Merlin υπάρχουν πέντε διαφορετικοί τρόποι υπολογισμού των μερικών παραγώγων της συνάρτησης $g_i = \frac{\partial f}{\partial x_i}$.

`FAST` Κάθε μία από τις παραγώγους $g_i$ προσεγγίζεται αριθμητικά από ένα τύπο εμπρόσθιων διαφορών δαπανώντας 2 κλήσεις της αντικειμενικής συνάρτησης.

`QUAD` Κάθε μία από τις παραγώγους $g_i$ προσεγγίζεται αριθμητικά από ένα τύπο κεντρικών διαφορών δαπανώντας 4 κλήσεις της συνάρτησης.

`NUMER` Κάθε μία από τις παραγώγους $g_i$ προσεγγίζεται αριθμητικά από ένα τύπο κεντρικών διαφορών δαπανώντας τουλάχιστον 6 κλήσεις της συνάρτησης.

`ANAL` Ολόκληρο το διάνυσμα **g** των πρώτων παραγώγων υπολογίζεται αναλυτικά από το υποπρόγραμμα granal που έχει προετοιμάσει ο χρήστης.

`MIXED` Κάθε μία από τις παραγώγους $g_i$ υπολογίζεται με ένα διαφορετικό τρόπο. Αυτό είναι χρήσιμο στην περίπτωση που δεν υπάρχουν αναλυτικοί τύποι για όλες τις παραγώγους $g_i$. Η εντολή `MIXED` αναμένει είσοδο μιάς γραμμής στη μορφή:

`δείκτης τρόπος_υπολογισμού δείκτης τρόπος_υπολογισμού ...`

Πχ.

```
1 anal 2 numer
```

ή χάριν συντομίας

```
1 a 2 n
```

όπου καθορίζεται ότι n $g_1$ θα υπολογιστεί αναλυτικά από το υποπρόγραμμα granal του χρήστη, ενώ n $g_2$ θα προσεγγιστεί με τον τρόπο `NUMER`.

## 5.2 Έλεγχος των παραγώγων Εντολές GERROR, GRADDIS και GRADCHECK

Κάθε ένας από τους αριθμητικούς τρόπους προσέγγισης των παραγώγων επιστρέφει μιά ποσοστιαία εκτίμηση του σφάλματος του υπολογισμού. Το αποδεκτό ποσοστό σφάλματος ελέγχεται με την εντολή `GERROR`.

Οι τιμές των παραγώγων $g_i$ για τις τρέχουσες τιμές των παραμέτρων $x_i$ μπορούν να ελεγθούν με την εντολή `GRADDIS`. Τυπικό παράδειγμα χρήσης της `GRADDIS` φαίνεται στον πίνακα 14, όπου n $g_1$ υπολογίζεται με τον τρόπο `NUMER` και n $g_2$ με τον τρόπο `ANAL`. Επιπλέον στα αριστερά των τιμών

των παραγώγων, εμφανίζεται η ένδειξη 0 ή -1 που σημαίνει ότι η παράγωγος υπολογίστηκε / δεν υπολογίστηκε με το επιτρεπτό ποσοστό σφάλματος.
Σύγκριση του τρόπου υπολογισμού των παραγώγων και κυρίως επιβεβαίωση ότι οι αναλυτικές παράγωγοι έχουν προγραμματιστεί σωστά, μπορεί να γίνει με την εντολή GRADCHECK. Τυπικό παράδειγμα χρήσης της GRADCHECK φαίνεται στον πίνακα 15.

GERROR Θέτει την τιμή του επιτρεπτού ποσοστιαίου σφάλματος κατά την αριθμητική προσέγγιση των παραγώγων. Η προκαθορισμένη τιμή είναι 1.E-4.

GRADDIS Εμφανίζει τις τιμές των παραγώγων για τον τρέχοντα τρόπο υπολογισμού.

GRADCHECK Ελέγχει τον υπολογισμό των παραγώγων με τον τρέχοντα τρόπο, συγκριτικά με κάποιο άλλο τρόπο υπολογισμου (που ζητείται ως είσοδος από το χρήστη).

Το Merlin περιέχει ενσωματωμένο μηχανισμό που αποτρέπει άσκοπους επαναληπτικούς υπολογισμούς των παραγώγων όταν αυτές απαιτούνται στο ίδιο σημείο. Πχ. Από τρείς διαδοχικές εντολές GRADDIS, μόνο η πρώτη θα υπολόγιζε στην πραγματικότητα τις παραγώγους.

Πίνακας 14: Παράδειγμα χρήσης της εντολής GRADDIS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
graddis  <cr>
 GRADDIS
   1)  RHO          2.49840000E+03               ANAL
   2)  SIGMA       -8.92000000E+02      0        NUMER


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

Πίνακας 15: Δείγμα από τη χρήση της GRADCHECK.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
gradcheck  <cr>
 GRADCHECK
ENTER DERIVATIVE MODE
anal  <cr>
 ------------------------------------------------------------------------------
                        ANAL                NUMER              ERROR %
 ------------------------------------------------------------------------------
   1)               2.49840E+03         2.49840E+03    0    8.427E-12
   2)              -8.92000E+02        -8.92000E+02    0    2.206E-11


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 5.3 Αυτόματη αλλαγή τρόπου παραγώγισης Η εντολή MAD

Κατά τη χρήση των ελαχιστοποιητών BFGS, DFP, CONGRA και TOLMIN που απαιτούν τον υπολογισμό των πρώτων παραγώγων της συνάρτησης, έχει παρατηρηθεί ότι όταν ο ελαχιστοποιητής βρίσκεται πολύ μακριά από το ελάχιστο, χονδρικές προσεγγίσεις των παραγώγων είναι αρκετές. Καθώς η μέθοδος πλησιάζει στο ελάχιστο, απαιτούνται προσεγγίσεις των παραγώγων υψηλότερης ακρίβειας. Το Merlin περιέχει ενσωματωμένο μηχανισμό που του επιτρέπει να αλλάζει αυτόματα τον τρόπο υπολογισμού των παραγώγων κάτω από ορισμένα κριτήρια.

Συγκεκριμένα ο τρόπος υπολογισμού της παραγώγου $g_i$ αλλάζει σε ακριβέστερο (από FAST σε QUAD ή από QUAD σε NUMER), όταν συντρέχει ένας από τους δύο λόγους:

α) Η εκτίμηση του σφάλματος για τον τρέχοντα τρόπο υπολογισμού είναι μεγαλύτερη από μιά προκαθορισμένη τιμή (TEHIGH).

β) Η απόλυτη τιμή της παραγώγου $g_i$ είναι μικρότερη από ένα προκαθορισμένο όριο (FQTHRE, QNTHRE).

Ο τρόπος υπολογισμού της παραγώγου $g_i$ αλλάζεται σε έναν λιγότερο ακριβή (από QUAD σε FAST ή από NUMER σε QUAD), όταν συντρέχει ένας από τους δύο λόγους:

α) Η εκτίμηση του σφάλματος για τον τρέχοντα τρόπο υπολογισμού είναι μικρότερη από μιά προκαθορισμένη τιμή (TELOW).

β) Η απόλυτη τιμή της παραγώγου $g_i$ είναι μεγαλύτερη από ένα προκαθορισμένο όριο (FQTHRE, QNTHRE).

Οι ανωτέρω παράμετροι ελέγχονται από την εντολή MAD. Ο πίνακας ελέγχου της εντολής MAD περιέχει τις παραμέτρους:

| | |
|---|---|
| MINIMUM TAYLOR ERROR | Το ελάχιστο επιτρεπτό σφάλμα (TELOW) κατά τον υπολογισμό των παραγώγων. Όταν το σφάλμα πέσει κάτω από αυτή την τιμή, ο τρόπος υπολογισμού αλλάζει σε ένα λιγότερο ακριβή. |
| MAXIMUM TAYLOR ERROR | Το μέγιστο επιτρεπτό σφάλμα (TEHIGH) κατά τον υπολογισμό των παραγώγων. Όταν το σφάλμα υπερβεί αυτή την τιμή, ο τρόπος υπολογισμού αλλάζει σε ένα ακριβέστερο. |
| FAST <==> QUAD THRESHOLD | Το όριο (FQTHRE) στην τιμή της παραγώγου για αλλαγή από FAST σε QUAD. |
| QUAD <==> NUMER THRESHOLD | Το όριο (QNTHRE) στην τιμή της παραγώγου για αλλαγή από QUAD σε NUMER. |
| MAD PRINTOUT (OFF/ON) | Καθορίζει αν θα υπάρχει έξοδος που θα πληροφορεί το χρήστη για την αλλαγή του τρόπου παραγώγισης. Επιτρεπτές τιμές είναι:<br>0 Δεν παράγεται έξοδος. |

1 Παράγεται έξοδος που πληροφορεί το χρήστη για την αλλαγή του τρόπου παραγώγισης και για το λόγο της αλλαγής.

Για να ενεργοποιηθεί ο αυτόματος τρόπος αλλαγής παραγώγων από τις μεθόδους ελαχιστοποίησης, πρέπει στους αντίστοιχους πίνακες ελέγχου να δοθεί η τιμή 1 στην παράμετρο `GRADIENT MODE` (βλέπε παράγραφο 10.3). Μετά το πέρας κάθε ελαχιστοποίησης που έχει γίνει με την αυτόματη αλλαγή παραγώγων, ο τρόπος υπολογισμού των παραγώγων τίθεται σε `MIXED`. Έχει βρεθεί ότι η αυτόματη αλλαγή παραγώγων επιταχύνει τους ελαχιστοποιητές, ειδικά στις περιπτώσεις συναρτήσεων με μεγάλο αριθμό μεταβλητών.

# Κεφάλαιο 6

# Εντολές ελέγχου

## 6.1 Έλεγχος του τρόπου λειτουργίας του Merlin Εντολές MODEDIS και LIMITS

Οι παράμετροι που επηρρεάζουν τη λειτουργία του Merlin μπορούν να ελεγθούν με τη χρήση της εντολής MODEDIS. Η εντολή LIMITS παρουσιάζει τις μέγιστες τιμές εσωτερικών παραμέτρων του προγράμματος και είναι χρήσιμη, όταν πραγματοποιούνται επεκτάσεις (βλέπε παράγραφο 13.1). Τυπικό αποτέλεσμα από τη χρήση των εντολών φαίνεται στον πίνακα 16.

Πίνακας 16: Αποτέλεσμα από χρήση των εντολών MODEDIS και LIMITS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
modedis  <cr>
 MODEDIS
 CURRENT OPERATION MODES
 ------------------------------------------------------------------------------
 NUMBER OF VARIABLES:    2                  LARGEST REAL:          1.00E+038
 NUMBER OF TERMS:        2                  SMALLEST NONZERO REAL: 1.00E-036
 BACKUP MODE:             NOBACK            MACHINE'S ACCURACY:    1.00E-15
 DERIVATIVE MODE:         NUMER             TARGET VALUE:           - NONE -
 PRINTOUT MODE:           FULLPRINT
 VARIABLES ARE CALLED BY: INDEX
 PROCESSING MODE:         IAF
 PANEL MODE:              PANELON
 CHARACTER SET:           ASCII
 LOOK ON DISK FOR MACROS: YES
 PROLOG COMMAND:           - NONE -
 EPILOG COMMAND:           - NONE -
 DATA FILE MODE:          DATASKIP
 DISPO FILE MODE:         DISPOSKIP
 FUNCTION EVALUATION:     ON
 FUNCTIONAL FORM:         GENERAL
 ------------------------------------------------------------------------------


     //////     MERLIN   IS AT YOUR COMMAND !!!
limits  <cr>
 LIMITS
 CURRENT LIMITS ARE:
 -------------------------------------------------
 NUMBER OF VARIABLES:                        200
 TERMS IN THE SUM OF SQUARES:                200
 NUMBER OF ALIASES:                           20
 MCL MEMORY:                                2500
 MCL STACK:                                  500
 NUMBER OF MCL FILES:                         20
 MACRO NESTING LEVEL:                         20
 NUMBER OF NUMERICAL FLAGS:                   20
 NUMBER OF CHARACTER FLAGS:                   10
 LENGTH OF A CHARACTER FLAG:                  30


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 6.2 Μορφή της αντικειμενικής συνάρτησης Εντολές GENERAL και SOS

Το Merlin αναγνωρίζει δύο μορφές αντικειμενικών συναρτήσεων, που επιλέγονται με αντίστοιχες εντολές:

`GENERAL` Καθορίζει ότι η αντικειμενική συνάρτηση είναι μία γενική μη γραμμική συνάρτηση. Για τον υπολογισμό της ο χρήστης πρέπει να έχει προετοιμάσει τον πηγαίο κώδικα Fortran για την `FUNTION FUNMIN` (βλέπε παράγραφο 3.1). Αυτή είναι η προκαθορισμένη μορφή συνάρτησης.

`SOS` Καθορίζει ότι η αντικειμενική συνάρτηση είναι άθροισμα τετραγώνων όπως περιγράφεται στην παράγραφο 3.3. Για τον υπολογισμό της ο χρήστης πρέπει να έχει προετοιμάσει τον πηγαίο κώδικα Fortran για την `SUBROUTINE SUBSUM` (βλέπε παράγραφο 3.3).

Και στις δύο ανωτέρω περιπτώσεις, εάν η παράγωγοι της συνάρτησης ειναι αναλυτικά γνωστές θα πρέπει να προγραμματιστούν ως `SUBROUTINE GRANAL` (βλέπε παράγραφο 3.2). Μόνο μία από τις `FUNCTION FUNMIN` ή `SUBROUTINE SUBSUM` πρέπει να προγραμματιστεί.

## 6.3 Τρόπος λειτουργίας. Εντολές IAF και BATCH

Το Merlin είναι ένα διαλογικό πρόγραμμα που όμως μπορεί να χρησιμοποιηθεί σε μη διαλογικό τρόπο λειτουργίας batch χωρίς την παρέμβαση του χρήστη. Αυτό μπορεί να γίνει είτε με την χρήση των οδηγιών διαμόρφωσης `INPUT_FILE` και `OUTPUT_FILE` στο αρχείο `CONFIG` (βλέπε παραγράφους 12.6, 12.7), ή με την ανακατεύθυνση των μονάδων εισόδου/εξόδου όταν το λειτουργικό σύστημα το επιτρέπει. Πχ. για το λειτουργικό σύστημα unix:

```
merlin < input_file > output_file
```

Στην περίπτωση που κατά τη διάρκεια της εκτέλεσης συμβεί κάποιο λάθος (πχ. υπάρχει μιά λανθασμένη εντολή στο αρχείο εισόδου), οι ενέργειες του Merlin καθορίζονται με τις εντολές:

`IAF` Στην περίπτωση λαθών το Merlin απλώς τυπώνει το προτρεπτικό σήμα και αναμένει την επόμενη εντολή. Στην περίπτωση εισόδου από αρχείο, διαβάζεται από το αρχείο η επόμενη εντολή. Αυτός είναι ο προκαθορισμένος τρόπος λειτουργίας.

`BATCH` Στην περίπτωση λαθών, το Merlin τερματίζει τη λειτουργία του τυπώνοντας κατάλληλα διαγνωστικά μηνύματα.

Ο καθορισμός του διαλογικού ή μη τρόπου λειτουργίας γίνεται είτε με τις ανωτέρω εντολές ή με την οδηγία διαμόρφωσης `MODE` στο αρχείο `CONFIG` (βλέπε παράγραφο 12.17). Εάν δεν δοθεί η οδηγία διαμόρφωσης `MODE`, κατά την εκκίνηση το Merlin ερωτά διαλογικά το χρήστη γιά τον επιθυμητό τρόπο λειτουργίας.

## 6.4 Έλεγχος εξόδου
## Εντολές FULLPRINT, HALFPRINT και NOPRINT

Το ποσό της εξόδου που παράγεται από το Merlin, ελέγχεται με τις εντολές:

`FULLPRINT` Όλα τα διαγνωστικά μηνύματα και οι προτροπές του Merlin εμφανίζονται. Αυτή είναι η προκαθορισμένη συμπεριφορά.

`HALFPRINT` Μόνο τα πιό σημαντικά διαγνωστικά μηνύματα και προτροπές εμφανίζονται.

`NOPRINT` Δεν παράγεται έξοδος.

Η προκαθορισμένη συμπεριφορά μπορεί να αλλάξει είτε με κάποια από τις ανωτέρω εντολές, ή με την οδηγία διαμόρφωσης `PRINTOUT` στο αρχείο `CONFIG` (βλέπε παράγραφο 12.14).

## 6.5 Μετρητές κλήσεων. Εντολή RESET

Το Merlin περιέχει ενσωματωμένους τέσερεις μετρητές οι τιμές των οποίων εμφανίζονται με χρήση των εντολών `SHORTDIS` και `VALDIS`:

α) Ο αριθμός τον κλήσεων της αντικειμενικής συνάρτησης από την αρχή της λειτουργίας του Merlin.
β) Ο αριθμός τον κλήσεων της αντικειμενικής συνάρτησης από την πιό πρόσφατη εντολή `RESET`.
γ) Ο αριθμός τον κλήσεων του υποπρογράμματος granal που υπολογίζει αναλυτικά τις παραγώγους, από την αρχή της λειτουργίας του Merlin.
δ) Ο αριθμός τον κλήσεων του υποπρογράμματος granal που υπολογίζει αναλυτικά τις παραγώγους, από την πιό πρόσφατη εντολή `RESET`.

Η εντολή `RESET` μηδενίζει τους μετρητές β και δ επιτρέποντας την καταμέτρηση επιμέρους κλήσεων από διάφορες μεθόδους ελαχιστοποίησης.

## 6.6 Παρουσίαση των πινάκων ελέγχου
## Εντολές PANELON και PANELOFF

Η εμφάνιση των πινάκων ελέγχου για τις εντολές ελαχιστοποίησης (αλλά και άλλες εντολές που χρησιμοποιούν πίνακες ελέγχου) καθορίζεται από τις εντολές:

`PANELON` Οι πίνακες ελέγχου εμφανίζονται και ο χρήστης μπορεί να εισάγει αλλαγές στις παραμέτρους ελέγχου. Αυτή είναι η προκαθορισμένη συμπεριφορά.

`PANELOFF` Οι πίνακες ελέγχου δεν εμφανίζονται, αλλά η μέθοδος προχωρεί με τις ήδη υπάρχουσες τιμές των παραμέτρων ελέγχου.

Σημειώστε ότι αν έχει δοθεί η εντολή NOPRINT, δεν παράγεται έξοδος, αλλά ο πίνακας ελέγχου εξακολουθεί να αναμένει την είσοδο αλλαγών, όπως περιγράφεται στην παράγραφο 4.5.

## 6.7 Μετονομασίες εντολών Εντολές ALIAS, UNALIAS και ALIASDIS

Είναι δυνατόν να καθοριστούν δεύτερα ονόματα γιά οποιαδήποτε από τις εντολές του Merlin. Τα δεύτερα ονόματα έχουν μήκος μέχρι 5 χαρακτήρες και ελέγχονται από τις εντολές:

| | |
|---|---|
| ALIAS | Καθορίζει το δεύτερο όνομα μιας εντολής. |
| UNALIAS | Ακυρώνει το δεύτερο όνομα μιας εντολής. |
| ALIASDIS | Εμφανίζει τα ήδη καθορισμένα δεύτερα ονόματα. |

Τυπικό παράδειγμα των ανωτέρω εντολών όπου το GD ορίζεται σαν δεύτερο όνομα της εντολής GRADDIS φαίνεται στον πίνακα 17.

Πίνακας 17: Χρήση των εντολών ALIAS, ALIASDIS και UNALIAS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
alias  <cr>
 ALIAS
 ENTER ALIAS
gd  <cr>
 ENTER ACTUAL COMMAND
graddis  <cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
aliasdis  <cr>
 ALIASDIS
 ALIAS        COMMAND
 -----        -------
 GD           GRADDIS


     //////     MERLIN   IS AT YOUR COMMAND !!!
unalias  <cr>
 UNALIAS
 ENTER ALIAS TO DELETE
gd


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 6.8 Απαγόρευση υπολογισμού της συνάρτησης Εντολές NOEVAL και EVALUATE

Κάθε φορά που γίνεται χρήση των εντολών POINT, MARGIN κα. το Merlin καλεί την αντικειμενική συνάρτηση. Πολλές φορές είναι επιθυμητό (ιδιαίτερα μέσω προγραμμάτων MCL, βλέπε κεφάλαιο 11) να αλλάξουμε τις τιμές πολλών παραμέτρων, ορίων κλπ., υπολογίζοντας τη συνάρτηση μία μόνο φορά στο τέλος, αφού ολοκληρώσουμε τις αλλαγές. Αυτό επιτυγχάνεται με τις εντολές:

NOEVAL Η αντικειμενική συνάρτηση δεν υπολογίζεται (δεν καλείται η FUNCTION FUNMIN ή η SUBROUTINE SUBSUM). Όλες οι εντολές του Merlin που απαιτούν υπολογισμό της αντικειμενικής συνάρτησης (πχ. εντολές ελαχιστοποίησης) δεν μπορούν να εκτελεστούν.

EVALUATE Η αντικειμενική συνάρτηση υπολογίζεται όποτε χρειαστεί. Αυτός είναι ο προκαθορισμένος τρόπος λειτουργίας. Επιπλέον, κάθε φορά που δίνεται η εντολή EVALUATE το Merlin εξαναγκάζεται να κάνει μία κλήση της αντικειμενικής συνάρτησης.

## 6.9 Ανάθεση συμβολικών ονομάτων Οι εντολές GODFATHER, NAME και INDEX

Η ανάθεση συμβολικών ονομάτων στις παραμέτρους της αντικειμενικής συνάρτησης γίνεται με χρήση της εντολής GODFATHER. Η εντολή GODFATHER αναμένει είσοδο μιάς γραμμής της μορφής:

```
δείκτης όνομα δείκτης όνομα ...
```

Πχ.

```
1 RHO 2 SIGMA
```

Τα συμβολικά ονόματα έχουν μήκος μέχρι 10 χαρακτήρες και η επιλογή τους αφήνεται στην απόλυτη διάθεση του χρήστη. Ο τρόπος αναφοράς στις παραμέτρους της συνάρτησης ελέχγεται από τις εντολές (βλέπε και παράγραφο 4.2):

NAME Καθορίζει ότι σε όλες τις εντολές που απαιτούν είσοδο μιάς γραμμής, οι παράμετροι της συνάρτησης θα αναφέρονται με τα συμβολικά τους ονόματα. Εάν σε κάποια παράμετρο δεν έχει δοθεί συμβολικό συμβολικό όνομα, τότε αυτή δεν θα μπορεί να αναφερθεί με τον τρόπο NAME.

INDEX Καθορίζει ότι οι παράμετροι θα αναφέρονται με τους δείκτες τους. Αυτή είναι η προκαθορισμένη συμπεριφορά.

## 6.10 Καθορισμός του συνόλου χαρακτήρων Η εντολή CSET

Όπως έχει αναφερθεί στην παράγραφο 2.2, το Merlin υποστηρίζει δύο σύνολα χαρακτήρων: ASCII και EBCDIC. Το σύνολο χαρακτήρων του υπολογιστή πρέπει να καθοριστεί με οδηγίες διαμόρφωσης (βλέπε παραγράφους 2.2 και 12.13) αλλά είναι δυνατόν να αλλάξει με την εντολή CSET. Εκτός από τα ASCII και EBCDIC, ο χρήστης μπορεί να εισάγει την έκφραση NORMAL, δηλώνοντας ότι το σύνολο χαρακτήρων δεν ανήκει σε καμία από τις δύο προηγούμενες κατηγορίες. Σε αυτή την περίπτωση όλη η είσοδος του Merlin πρέπει να δοθεί με κεφαλαίους χαρακτήρες. Το προκαθορισμένο σύνολο χαρακτήρων είναι ASCII.

## 6.11 Ανακατεύθυνση της εξόδου Εντολές HIDEOUT και REVEAL

Προσωρινή ανακατεύθυνση της εξόδου του Merlin μπορεί να γίνει με χρήση των εντολών:

HIDEOUT Ζητείται ένας ακέραιος αριθμός nnn, ο οποίος θα προσδιορίσει το όνομα του αρχείου που θα δεχτεί προσωρινά την έξοδο του Merlin. Το αρχείο που θα δημιουργηθεί είναι το HIDEnnn. Από τη χρήση της εντολής HIDEOUT, μέχρι τη χρήση της εντολής REVEAL, όλη η έξοδος του Merlin γράφεται στο αρχείο HIDEnnn.

REVEAL Επαναφέρει την έξοδο του Merlin στον προκαθορισμένο προορισμό αναιρώντας τη λειτουργία της προηγούμενης εντολής HIDEOUT.

## 6.12 Έλεγχος βημάτων Εντολές STEP, STEPDIS, STEPALL και ADJUST

Οι ελαχιστοποιητές ROLL και RANDOM γιά κάθε παράμετρο $x_i$ χρησιμοποιούν (και τροποποιούν) τα βήματα $s_i$ (βλέπε παράγραφο 10.7, 10.8). Οι παρακάτω εντολές σχετίζονται με τον έλεγχο των βημάτων $s_i$.

STEP Ζητείται είσοδος μιάς γραμμής με τις τιμές των βημάτων. Εάν αυτές δεν καθοριστούν με χρήση της εντολής STEP, η προκαθορισμένη τιμή τους είναι 1.

STEPDIS Εμφανίζονται οι τρέχουσες τιμές των βημάτων $s_i$.

STEPALL Καθορίζονται τα βήματα γιά όλες τις παραμέτρους $x_i$, με μιά γραμμική αναζήτηση στη διεύθυνση που δείχνουν οι παράγωγοι της συνάρτησης.

ADJUST Καθορίζονται τα βήματα γιά όλες τις παραμέτρους $x_i$, χωρίς να υπολογιστούν οι παράγωγοι της συνάρτησης.

Τυπικός διάλογος χρήσης των εντολών STEP και STEPDIS φαίνεται στον πίνακα 18.

Πίνακας 18: Χρήση των εντολών STEP και STEPDIS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
step  <cr>
 STEP
1  1.5  <cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
stepdis  <cr>
 STEPDIS
      1)   ....  1.500
      2)   ....  1.000


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

# Κεφάλαιο 7

# Άλλες δυνατότητες

## 7.1 Όρια στις παραμέτρους Εντολές MARGIN και DEMARGIN

Όρια του τύπου:

$$a_i \leq x_i \leq b_i \qquad i = 1, 2, \ldots N$$

μπορεί να αντιμετωπιστούν από τους ελαχιστοποιητές του Merlin. Κάθε μία από τις παραμέτρους $x_i$ μπορεί να έχει αριστερό όριο $a_i$ , δεξί όριο $b_i$ , ή και τα δύο. Τα όρια τίθενται με την εντολή MARGIN. Αρχικά ζητείται είσοδος μιάς γραμμής με τα αριτερά όρια $a_i$ και αμέσως μετά είσοδος μιάς γραμμής με τα δεξιά όρια $b_i$. Τυπικό παράδειγμα της εντολής MARGIN φαίνεται στον πίνακα 19. Οι τιμές των ορίων εμφανίζονται με χρήση της εντολής SHORTDIS.

Ακύρωση των ορίων γίνεται με την εντολή DEMARGIN. Αρχικά ζητείται είσοδος μιας γραμμής για τις παραμέτρους των οποίων επιθυμούμε να ακυρώσουμε το αριστερό όριο. Αμέσως μετά ζητείται n αντίστοιχη είσοδος μιάς γραμμής για τις παραμέτρους των οποίων επιθυμούμε να ακυρώσουμε το δεξί όριο. Τυπικό παράδειγμα φαίνεται στον πίνακα 20.

Πίνακας 19: Παράδειγμα χρήσης της εντολής MARGIN καθώς και εξέταση των ορίων αμέσως μετά με την εντολή SHORTDIS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
margin  <cr>
 MARGIN
                    ENTER LEFT  MARGINS SEQUENCE
1 -10 2 -6
                    ENTER RIGHT MARGINS SEQUENCE
2 14


     //////     MERLIN   IS AT YOUR COMMAND !!!
shortdis  <cr>
 SHORTDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                            1               0
 SINCE LAST RESET:                 1               0
 ------------------------------------------------------------------------------
   1)           .......   1.4000000000000         -10.00
   2)           .......  -2.5000000000000         -6.000        14.00
         VALUE.......     1989.3200000000


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

Πίνακας 20: Παράδειγμα χρήσης της εντολής DEMARGIN καθώς και εξέταση των ορίων αμέσως μετά με την εντολή SHORTDIS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
demargin  <cr>
 DEMARGIN
                    ENTER LEFT  MARGINS SEQUENCE
1  2  <cr>
                    ENTER RIGHT MARGINS SEQUENCE
2  <cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
shortdis  <cr>
 SHORTDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                            1               0
 SINCE LAST RESET:                 1               0
 ------------------------------------------------------------------------------
   1)           .......   1.4000000000000
   2)           .......  -2.5000000000000
         VALUE.......     1989.3200000000


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 7.2 Σταθεροποίηση των παραμέτρων Εντολές FIX, LOOSE, FIXALL και LOOSALL

Πολλές φορές είναι επιθυμητό να σταθεροποιήσουμε την τιμή ορισμένων παραμέτρων σε κάποια συγκεκριμένη τιμή και να προχωρήσουμε σε ελαχιστοποίηση χρησιμοποιώντας τις υπόλοιπες. Η σταθεροποίηση/αποσταθεροποίηση των παραμέτρων ελέγχεται από τις εντολές:

FIX Ζητείται είσοδος μιάς γραμμής για να καθοριστούν οι παράμετροι προς σταθεροποίηση. Οι παράμετροι που προσδιορίζουμε σταθεροποιούνται στην τρέχουσα τιμή τους ενώ με τη χρήση της SHORTDIS εμφανίζεται η ένδειξη `..  *  ..` αριστερά της τιμής τους.

LOOSE Ακυρώνει τη λειτουργία της εντολής FIX. Οι παράμετροι γιά τις οποίες ακυρώνεται η λειτουργία της σταθεροποίησης δίνονται ως είσοδος μιάς γραμμής.

LOOSALL Ακυρώνει τη σταθεροποίηση όλων των παραμέτρων.

FIXALL Σταθεροποιεί όλες τις παραμέτρους στην τρέχουσα τιμή τους. Προφανώς πρέπει να προηγηθεί εντολή LOOSE πριν από οποιαδήποτε ελαχιστοποίηση, ειδάλως κατάλληλα διαγνωστικά μηνύματα θα δείξουν ότι όλες οι παράμετροι είναι καθηλωμένες.

Τυπικά παραδείγματα χρήσης των εντολών FIX και LOOSE φαίνονται στους πίνακες 21 και 22.

Πίνακας 21: Παράδειγμα χρήσης της εντολής FIX καθώς και εξέταση των παραμέτρων αμέσως μετά με την εντολή SHORTDIS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
fix  <cr>
 FIX
1  <cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
shortdis  <cr>
 SHORTDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                            1               0
 SINCE LAST RESET:                 1               0
 ------------------------------------------------------------------------------
   1)           .. * ..   1.4000000000000
   2)           .......  -2.5000000000000
         VALUE.......     1989.3200000000


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

Πίνακας 22: Παράδειγμα χρήσης της εντολής LOOSE καθώς και εξέταση των παραμέτρων αμέσως μετά με την εντολή SHORTDIS.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
loose  <cr>
 LOOSE
1  <cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
shortdis  <cr>
 SHORTDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                            1               0
 SINCE LAST RESET:                 1               0
 ------------------------------------------------------------------------------
   1)           .......   1.4000000000000
   2)           .......  -2.5000000000000
         VALUE.......     1989.3200000000


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 7.3 Συσσώρευση αρχικών σημείων. Η εντολή ACCUM

Τυχαία αρχικά σημεία εκκίνησης για τους ελαχιστοποιητές μπορούν να παραχθούν με την εντολή ACCUM. Για το σκοπό αυτό κάθε παράμετρος $x_i$ θα πρέπει είτε να έχει άνω και κάτω όριο (μετά από χρήση της εντολής MARGIN), ή η τιμή της να έχει σταθεροποιηθεί (με την εντολή FIX). Η ACCUM παράγει τυχαία δοκιμαστικά σημεία, και συσσωρεύει στο αρχείο INIPO (βλέπε παράγραφο 7.3) όσα δίνουν τιμή συνάρτησης μικρότερη από μιά τιμή στόχο. Η ACCUM χρησιμοποιεί ένα πίνακα ελέγχου για την εισαγωγή των απαραίτητων στοιχείων.

Οι παράμετροι του πίνακα ελέγχου της ACCUM είναι:

| | |
|---|---|
| NUMBER OF CALLS | Μέγιστος επιτρεπτός αριθμός κλήσεων της αντικειμενικής συνάρτησης. |
| NUMBER OF POINTS | Ζητούμενος αριθμός σημείων με τιμή μικρότερη της τιμής-στόχου. Κάθε τέτοιο σημείο προστίθεται στο αρχείο INIPO. |
| TARGET VALUE | Η τιμή-στόχος με την οποία συγκρίνονται τα δοκιμαστικά σημεία που δειγματίζει η ACCUM. |
| CANCEL-BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |

Η λειτουργία της ACCUM τερματίζεται όταν είτε ο μέγιστος επιτρεπτός αριθμός κλήσεων εξαντληθεί, ή βρεθούν τόσα σημεία όσα ζητούνται με την παράμετρο NUMBER OF POINTS. Μετά το τέλος της ACCUM, και αφού έχουν συσσωρευθεί σημεία στο αρχείο INIPO, ο χρήστης μπορεί να επιθεωρήσει το αρχείο με την εντολή INSPECT και να επιλέξει οποιοδήποτε από τα σημεία με την εντολή PICK (βλέπε παράγραφο 8.6). Τυπικό παράδειγμα φαίνεται στον πίνακα 23.

Πίνακας 23: Δείγμα χρήσης της εντολής ACCUM.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
accum  <cr>
 ACCUM
 ------------------------------------------------------------------------------
                            ACCUM - PANEL
 ------------------------------------------------------------------------------
  INDEX    DESCRIPTION                 VALUE       MENU
    1)   NUMBER OF CALLS                300        ANY INTEGER
    2)   NUMBER OF POINTS                 1        ANY INTEGER
    3)   TARGET VALUE                     0.       ANY REAL
    4)   CANCEL-BUTTON                   1          ( 0/1/2 )
 ------------------------------------------------------------------------------
    ENTER CHANGES
1 1000 2 5 3 350
         NUMBER OF POINTS  APPENDED TO FILE INIPO:   5


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 7.4 Γραφική απεικόνιση. Η εντολή GRAPH

Γραφικές απεικονίσεις της συνάρτησης με τη χρήση χαρακτήρων μπορούν να γίνουν με την εντολή GRAPH. Η εντολή GRAPH χρησιμοποιεί ένα πίνακα ελέγχου για την εισαγωγή των απαραίτητων στοιχείων.

Οι παράμετροι του πίνακα ελέγχου της GRAPH είναι:

| | |
|---|---|
| VARIABLE INDEX | Ο δείκτης της παραμέτρου της αντικειμενικής συνάρτησης ως προς την οποία ζητείται γραφική απεικόνιση. Όλες οι υπόλοιπες παράμετροι θεωρούνται σταθερές. |
| NO OF POINTS | Ο αριθμός των σημείων (και κατά συνέπεια κλήσεων της συνάρτησης) που θα χρησιμοποιηθούν. |
| FROM | Μαζί με την παράμετρο TO καθορίζουν το διάστημα στο οποίο ζητείται η απεικόνιση. |
| TO | Μαζί με την παράμετρο FROM καθορίζουν το διάστημα στο οποίο ζητείται η απεικόνιση. |
| LINES | Καθορίζει τον αριθμό των γραμμών που θα καταλάβει η απεικόνιση στην οθόνη. |
| COLOUMNS | Καθορίζει τον αριθμό των στηλών που θα καταλάβει η απεικόνιση στην οθόνη. |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |

Τυπικό παράδειγμα χρήσης της GRAPH φαίνεται στον πίνακα 24.

Πίνακας 24: Τυπικό παράδειγμα χρήσης της GRAPH.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
graph  <cr>
 GRAPH
 ------------------------------------------------------------------------------
                          GRAPH  - PANEL
 ------------------------------------------------------------------------------
  INDEX    DESCRIPTION                VALUE       MENU
    1)    VARIABLE   INDEX             1          (1 -  2)
    2)    N0 OF POINTS                 30         ANY INTEGER
    3)    FROM                       -1.000E+05
    4)    TO                          1.000E+05
    5)     LINES                       17         ANY INTEGER
    6)    COLOUMNS                     64         (2-124)
    7)    CANCEL-BUTTON                1          ( 0/1/2 )
 ------------------------------------------------------------------------------
    ENTER CHANGES
1  2  3  -1.5  4  5  <cr>


             .+....+....+....+....+....+....+....+....+....+....+....+....+....
    1.600E+03+               .                                              *+
             .               .                                               .
 F  1.400E+03+               .                                            *  +
 U           .               .                                          *    .
 N  1.200E+03+               .                                               +
 C           .               .                                       *       .
 T  1.000E+03+               .                                     *         +
 I           .               .                                               .
 O   800.    +               .                                  *            +
 N           .               .                               *               .
     600.    +*              .                            *                  +
             .  *            .                         *                     .
 V   400.    +    *          .                      *                        +
 A           .       * *     .                   *                           .
 L   200.    +           *   .                * *                            +
 U           .             * *             ***                               .
 E  0.119    +               . *  * * X * * *                                 +
             .+....+....+....+....+....+....+....+....+....+....+....+....+....
       -1.500000E+00   I    5.634921E-01   I    2.626984E+00   I
                -4.682540E-01        1.595238E+00        3.658730E+00
                VARIABLE :   2)


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 7.5 Μαζική ανάθεση αρχικών τιμών. Η εντολή INIT

Η εντολή INIT επιτρέπει στο χρήστη να δώσει αρχικές τιμές στις μεταβλητές της συνάρτησης, διαβάζοντάς τις από κάποιο αρχείο. Οι αρχικές τιμές δεν πρέπει να είναι αποθηκευμένες στο αρχείο με κάποιο συγκεκριμένο τρόπο διότι η ανάγνωσή τους γίνεται με τη χρήση μιας free-format εντολής `read`. Επιπλέον είναι δυνατόν να διαβαστούν από αρχείο οι τιμές των ορίων, των βημάτων (βλέπε παράγραφο 6.12) και η κατάσταση σταθεροποίησης κάθε μεταβλητής (βλέπε παράγραφο 7.2). Σε αυτή την τελευταία περίπτωση, στο αρχείο πρέπει να υπάρχει ο αριθμός 0 ή 1 για μιά σταθεροποιημένη ή ελεύθερη μεταβλητή αντίστοιχα. Παράδειγμα χρήσης της εντολής `INIT` φαίνεται στον πίνακα 25.

Πίνακας 25: Ανάθεση αρχικών τιμών με την εντολή INIT.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
init  <cr>
 INIT
 ------------------------------------------------------------------------------
 :                               SELECTION MENU                                 :
 ------------------------------------------------------------------------------
 :                                      -1 ... FOR LEFT - MARGINS               :
 :                          ENTER :      0 ... FOR    POINT                     :
 :                                      +1 ... FOR RIGHT - MARGINS              :
 :                                       2 ... FOR FIX - STATUS                 :
 :                                       3 ... FOR STEPS                        :
 ------------------------------------------------------------------------------
                    ENTER SELECTION : >>
0  <cr>
 ...   ENTER FILE NAME   ...
start  <cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

# Κεφάλαιο 8

# Αρχεία

Το Merlin χρησιμοποιεί μία σειρά αρχείων με προκαθορισμένα ονόματα για ειδικούς σκοπούς. Στην περιγραφή που ακολουθεί, τα ονόματα των αρχείων αναφέρονται με κεφαλαίους χαρακτήρες, όπως είναι προγραμματισμένα στον πηγαίο κώδικα Fortran. Πρέπει να χρησιμοποιούνται με αυτό τον τρόπο, εάν το λειτουργικό σύστημα κάτω από το οποίο λειτουργεί το Merlin θεωρεί διαφορετικά ονόματα που εισάγονται με κεφαλαίους ή πεζούς χαρακτήρες (πχ. unix).

Στα αρχεία BACKUP, DATA, DISPO, STORE και INIPO αποθηκεύονται οι τιμές των παραμέτρων της αντικειμενικής συνάρτησης, οι τιμές των ορίων των τους, τα συμβολικά ονόματα που έχει επιλέξει ο χρήστης κα. Η μικρότερη μονάδα αποθήκευσης στα πέντε αυτά αρχεία είναι η "εγγραφή". Τυπικά περιεχόμενα μιάς εγγραφής φαίνονται στον πίνακα 26. Το αρχείο DATA περιέχει το πολύ μία εγγραφή, ενώ τα αρχεία BACKUP, DISPO, STORE και INIPO μπορεί να περιέχουν παραπάνω από μία εγγραφές.

Πίνακας 26: Τρόπος αποθήκευσης μιάς εγγραφής στα αρχεία BACKUP, DATA, DISPO, STORE και INIPO.

```
  1) RHO       .......   1.4000000000000        -10.00        10.00
  2) SIGMA     .......  -2.5000000000000        -40.00        40.00
        VALUE.......     1989.3200000000
```

## 8.1 Το αρχείο BACKUP
## Εντολές NOBACK, LASTBACK και FULLBACK

Το αρχείο BACKUP χρησιμεύει για την αποθήκευση εγγραφών μετά το πέρας κάθε μεθόδου ελαχιστοποίησης. Η αποθήκευση στο αρχείο BACKUP ελέγχεται από τις παρακάτω εντολές:

NOBACK Τίποτε δεν αποθηκεύεται στο αρχείο BACKUP. Αυτή είναι η προκαθορισμένη συμπεριφορά.

LASTBACK Οι τιμές των παραμέτρων της συνάρτησης αντικαθιστούν πλήρως τα περιεχόμενα του αρχείου. Με τον τρόπο αυτό το πιό πρόσφατο σύνολο παραμέτρων μένει κάθε φορά στο αρχείο.

FULLBACK Μετά το τέλος μιας εντολής ελαχιστοποίησης, μία εγγραφή προστίθεται στο τέλος του αρχείου. Το μέγεθος του αρχείου μεγαλώνει καθώς ο χρήστης εκτελεί εντολές ελαχιστοποίησης.

## 8.2 Το αρχείο DATA
## Εντολές DATASKIP και DATAWRITE

Το αρχείο DATA ενημερώνεται πρίν την εκτέλεση μιάς εντολής ελαχιστοποίησης, έτσι ώστε να περιέχει πληροφορίες για τις προηγούμενες και όχι τις τρέχουσες τιμές των παραμέτρων. Στο αρχείο DATA υπάρχει το πολύ μία εγγραφή. Η αποθήκευση στο αρχείο ελέγχεται από τις εντολές:

DATASKIP Στο αρχείο δεν αποθηκεύεται τίποτε. Αυτή είναι η προκαθορισμένη συμπεριφορά.
DATAWRITE Στο αρχείο αποθηκεύεται μία εγγραφή, όπως περιγράφεται παραπάνω.

## 8.3 Το αρχείο DISPO
## Εντολές DISPOSKIP και DISPOWRITE

Το αρχείο DISPO περιέχει πληροφορίες για σημεία που ο χρήστης επιθυμεί να απορρίψει προσωρινά και να δοκιμάσει άλλες αρχικές τιμές γιά την ελαχιστοποίηση. Ενημερώνεται όταν δοθεί μία από τις εντολές PICK, POINT, INIT ή ACCUM. Η αποθήκευση στο αρχείο ελέγχεται από τις εντολές:

DISPOSKIP Τίποτε δεν αποθηκεύεται στο αρχείο.
DISPOWRITE Νέες εγγραφές προστίθενται στο τέλος του αρχείου όταν δοθεί μία από τις εντολές PICK, POINT, INIT ή ACCUM.

## 8.4 Το αρχείο STORE. Η εντολή MEMO

Στο αρχείο STORE αποθηκεύονται σημεία επιλεγμένα από το χρήστη. Μία εγγραφή προστίθενται στο τέλος του αρχείου κάθε φορά που ο χρήστης δίνει την εντολή MEMO.

## 8.5 Το αρχείο INIPO

Στο αρχείο INIPO συσσωρεύονται σημεία που έχουν βρεθεί με την μέθοδο παραγωγής τυχαίων αρχικών σημείων ACCUM (βλέπε παράγραφο 7.3).

## 8.6 Επιθεώρηση των αρχείων
## Οι εντολές INSPECT και PICK

Ο χρήστης μπορεί να επιθεωρήσει τα αρχεία BACKUP, DATA, DISPO, STORE και INIPO με την εντολή INSPECT. Η εντολή INSPECT ζητάει το όνομα του αρχείου προς επιθεώρηση (που πρέπει να

είναι κάποιο από τα πέντε παραπάνω) και κατόπιν εμφανίζει μία προς μία τις εγγραφές του αρχείου. Τυπικό παράδειγμα επιθεώρησης αρχείου BACKUP που περιέχει δύο εγγραφές δίνεται στον πίνακα 27. Η επιθεώρηση μπορεί να διακοπεί σε οποιαδήποτε εγγραφή εισάγοντας κάποιον χαρακτήρα εκτός από <space> $ ή <cr>, όταν εμφανιστεί η προτροπή:

ENTER SPACE OR $ TO CONTINUE

Η τελευταία εγγραφή που επιθεωρήσαμε μπορεί να διαβαστεί από το Merlin και οι αποθηκευμένες τιμές να γίνουν τρέχουσες τιμές των παραμέτρων με τη χρήση της εντολής PICK. Η εντολή PICK πρέπει να δίνεται αμέσως μετά το πέρας μιας εντολής INSPECT.

Πίνακας 27: Χρήση της εντολής INSPECT. Το επιθεωρούμενο αρχείο BACKUP έχει 2 εγγραφές.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
inspect  <cr>
 INSPECT
ENTER FILE NAME TO INSPECT
BACKUP  <cr>
   1) RHO       .......  0.94418004886991
   2) SIGMA     .......  0.88425380540125
         VALUE.......    8.33182541468869E-03
ENTER SPACE OR $ TO CONTINUE
<cr>
   1) RHO       .......  -2.1233839198428         -10.00        10.00
   2) SIGMA     .......   4.5156161299641         -40.00        40.00
         VALUE.......     9.7602287621535
ENTER SPACE OR $ TO CONTINUE
<cr>


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 8.7 Η εντολή DISCARD

Τα περιεχόμενα οποιουδήποτε από τα αρχεία BACKUP, DATA, DISPO, STORE και INIPO μπορούν να καταστραφούν με την εντολή DISCARD. Η εντολή DISCARD αφού ζητήσει το όνομα του αρχείου, καταστρέφει τα περιεχόμενά του γράφοντας μία μοναδική εγγραφή με τις πληροφορίες του τρέχοντος σημείου.

## 8.8 Το αρχείο HELP
## Οι εντολές HELP, INTRODUCE και LIST

Το αρχείο HELP περιέχει τις πληροφορίες που εμφανίζονται όταν χρησιμοποιηθεί η εντολή HELP. Απόσπασμα του αρχείου φαίνεται στον πίνακα 28. Ο χρήστης μπορεί να επιμηκύνει το αρχείο μέχρι 1000 γραμμές, ενώ κείμενα βοήθειας για επεκτάσεις που έχουν προστεθεί από το χρήστη πρέπει να επισημαίνονται με την λέξη EXTENSION αντί γιά STANDARD.

Η εντολή INTRODUCE ανιχνεύει το αρχείο HELP και αναφέρει την ύπαρξη επεκτάσεων από το χρήστη, ενώ η LIST εμφανίζει ένα συνοπτικό κατάλογο με όλες τις εντολές Merlin.

Πίνακας 28: Απόσπασμα του αρχείου HELP.

```
***STANDARD***MACRO
INITIATES MACRO CONSTRUCTION. IT PROMPTS THE USER FOR A MACRO-NAME
AND CONSECUTIVELY FOR MERLIN COMMANDS. A MACRO - NAME MUST ALWAYS
START WITH A PERIOD AND SHOULD NOT BE LONGER THAN TEN CHARACTERS.
TO END A MACRO CONSTRUCTION SESSION ENTER THE COMMAND CLEAR.

***STANDARD***MEMO
APPENDS THE CURRENT POINT INFORMATION TO FILE STORE.

***STANDARD***CLEAR
INSTRUCTS MERLIN TO TERMINATE A MACRO - CONSTRUCTION SESSION.

***STANDARD***SIMPLEX
INVOKES THE SIMPLEX METHOD OF MINIMIZATION. PANEL PROMPTING.
```

## 8.9 Τα αρχεία MACROF και CONFIG

Σχετίζονται το πρώτο με τα παραμετρικά εντολοπακέτα και το δεύτερο με τη διαμόρφωση του συστήματος. Περιγράφονται αναλυτικά στα κεφάλαια 9 και 12 αντίστοιχα.

## 8.10 Η εντολή CATALOG

Ανάλογα με τους επιλεγμένους τρόπους ενημέρωσης των αρχείων (NOBACK, LASTBACK, DATAWRITE, DATASKIP κλπ.) που περιγράφηκαν παραπάνω, ορισμένα αρχεία θα περιέχουν εγγραφές, ενώ άλλα όχι. Η κατάσταση των αρχείων φαίνεται με την εντολή CATALOG, της οποίας τυπικό παράδειγμα φαίνεται στον πίνακα 29. Τα αρχεία που δεν περιέχουν εγγραφές έχουν την ένδειξη EMPTY.

Πίνακας 29: Αποτέλεσμα από τη χρήση της εντολής CATALOG.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
catalog  <cr>
 CATALOG

                    *** MERLIN - FILES ***
-------------------------------------------------------------------
  DATA         STORE        INIPO        DISPO        BACKUP       MACROF

  EMPTY        EMPTY        EMPTY        EMPTY          *            *
-------------------------------------------------------------------



     //////     MERLIN   IS AT YOUR COMMAND !!!
```

# Κεφάλαιο 9

# Παραμετρικά εντολοπακέτα

Το Merlin αναγνωρίζει και εκτελεί παραμετρικά εντολοπακέτα. Ένα εντολοπακέτο είναι μια συλλογή από εντολές Merlin και κλήσεις αλλών εντολοπακέτων αποθηκευμένων σε κάποιο αρχείο. Το όνομα του κάθε εντολοπακέτου δεν πρέπει να υπερβαίνει τους 10 χαρακτήρες.

## 9.1 Είδη εντολοπακέτων

Υπάρχουν δύο είδη εντολοπακέτων:

α) Εντολοπακέτα δίσκου.
Βρίσκονται το καθένα σε ένα ξεχωριστό αρχείο. Το όνομα του εντολοπακέτου ταυτίζεται με το όνομα του αρχείου στο οποίο είναι αποθηκευμένο. Τυπικό παράδειγμα τέτοιου εντολοπακέτου φαίνεται στον πίνακα 30. Κατασκευάζονται από το χρήστη με τη βοήθεια κάποιου συντάκτη κειμένου.

Πίνακας 30: Τυπικό παράδειγμα εντολοπακέτου δίσκου.

```
point
1 1.4 2 -2.5
<cr>
shortdis
graddis
```

β) Εντολοπακέτα βιβλιοθήκης.
Το Merlin διατηρεί ένα ειδικό αρχείο με το όνομα MACROF στο οποίο είναι δυνατό να συνυπάρξουν πολλά εντολοπακέτα μαζί. Τα εντολοπακέτα είναι αποθηκευμένα το ένα μετά το άλλο ώς εξής:

```
*
. όνομα_εντολοπακέτου
εντολή
εντολή
...
clear
```

Η εντολή `clear` χρησιμοποιείται για να δηλώσει το τέλος του εντολοπακέτου. Τυπικό παράδειγμα του αρχείου MACROF με δύο εντολοπακέτα φαίνεται στον πίνακα 31. Τα εντολοπακέτα του αρχείου MACROF μπορούν να κατασκευαστούν είτε με τον συντάκτη κειμένου που διαθέτει το λειτουργικό σύστημα, ή με την εντολή MACRO του Merlin.

Σε ένα εντολοπακέτο μπορούν να υπάρχουν κλήσεις άλλων εντολοπακέτων, αναδρομικές κλήσεις του ιδίου ή άλλων εντολοπακέτων δεν επιτρέπονται.

Πίνακας 31: Τυπικό παράδειγμα του αρχείου MACROF με δύο εντολοπακέτα.

```
*
.adis
anal
graddis
valdis
clear
*
.na
godfather
1 rho 2 sigma
shortdis
clear
```

## 9.2 Κατασκευή εντολοπακέτων. Η εντολή MACRO

Η εντολή MACRO βοηθά το χρήστη να κατασκευάσει ένα εντολοπακέτο και να το τοποθετήσει στο αρχείο MACROF. Αρχικά ζητείται το όνομα του εντολοπακέτου και κατόπιν οι εντολές που το αποτελούν. Ο τερματισμός της διαδικασίας γίνεται με την εισαγωγή της εντολής CLEAR. Τυπικό παράδειγμα κατασκευής εντολοπακέτου φαίνεται στον πίνακα 32.

Πίνακας 32: Κατασκευή εντολοπακέτου με την εντολή MACRO.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
macro  <cr>
 MACRO
** ENTER MACRO - NAME **
.na  <cr>
** MACRO EDITING STARTS **
** ENTER DESIRED COMMAND **
godfather  <cr>
** ENTER DESIRED COMMAND **
1 RHO 2 SIGMA <cr>
** ENTER DESIRED COMMAND **
shortdis  <cr>
** ENTER DESIRED COMMAND **
clear  <cr>
 ** MACRO EDITING COMPLETE **


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 9.3 Κλήση των εντολοπακέτων Οι εντολές DISKON, DISKOFF

Οι κλήσεις των εντολοπακέτων ξεχωρίζουν από τις εντολές Merlin από την αρχική τελεία. Τα εντολοπακέτα βιβλιοθήκης που βρίσκονται στο αρχείο MACROF είναι πάντα διαθέσιμα γιά χρήση. Η αναγνώριση των εντολοπακέτων δίσκου ελέγχεται από τις εντολές:

DISKON Επιτρέπει την αναγνώριση εντολοπακέτων δίσκου. Αυτή είναι η προκαθορισμένη συμπεριφορά.

DISKOFF Απαγορεύει την αναγνώριση εντολοπακέτων δίσκου.

Τυπικό παράδειγμα χρήσης εντολοπακέτου φαίνεται στον πίνακα 33.

Πίνακας 33: Κλήση εντολοπακέτου.

```
    //////     MERLIN   IS AT YOUR COMMAND !!!
.adis  <cr>
            ...  EXECUTION OF MACRO .ADIS HAS BEEN INITIATED  ...

 ANAL
 GRADDIS
   1)               2.49840000E+03               ANAL
   2)              -8.92000000E+02               ANAL
 VALDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                             1               1
 SINCE LAST RESET:                  1               1
 ------------------------------------------------------------------------------

                  VALUE =   1989.3200000000

 CLEAR
           ...  PROCESSING OF MACRO .ADIS HAS BEEN COMPLETED  ...



    //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 9.4 Παραμετρικά εντολοπακέτα

Τα εντολοπακέτα είναι δυνατόν να δέχονται ορίσματα κατά την κλήση τους. Τα ορίσματα αυτά αναφέρονται ως [1], [2] κλπ. μέσα στο εντολοπακέτο και αντικαθίστανται από τα πραγματικά ορίσματα που έχει εισάγει ο χρήστης κατά την κλήση του. Πχ. το παρακάτω εντολοπακέτο με όνομα .set θέτει ώς τιμή στην παράμετρο $x_2$ της συνάρτησης το πρώτο όρισμα που εισάγει ο χρήστης.

```
point
2 [1]
<cr>
shortdis
```

Παραδείγματα κλήσης του παραπάνω εντολοπακέτου είναι:

```
.set 7
.set -2.5
.set 68.23
```

Το σύμβολο `[0]` αντιπροσωπεύει το όνομα του εντολοπακέτου, ενώ το `[#]` τον αριθμό των ορισμάτων που έχει εισάγει ο χρήστης κατά την κλήση του εντολοπακέτου. Στο ανωτέρω παράδειγμα το `[0]` θα ήταν set, ενώ το `[#]` θα ήταν 1. Αν ο χρήστης εισάγει λιγότερα ορίσματα από αυτά που χρειάζεται το εντολοπακέτο, τότε η θέση των επιπλέον ορισμάτων μένει κενή. Ο αριθμός των ορισμάτων που μπορεί να έχει ένα εντολοπακέτο περιορίζεται μόνο από το μέγεθος της γραμμής εισόδου, που είναι 80 χαρακτήρες.

# Κεφάλαιο 10

# Οι αλγόριθμοι ελαχιστοποίησης

## 10.1 Συμβολισμός

Παρακάτω δίνεται μία περιληπτική περιγραφή των αλγορίθμων ελαχιστοποίησης του Merlin καθώς και η σημασία των παραμέτρων στους πίνακες ελέγχου. Τα μικρά έντονα γράμματα δηλώνουν διανύσματα, ενώ τα έντονα κεφαλαία δηλώνουν πίνακες. Επειδή όλες οι μέθοδοι είναι επαναληπτικές, οι ποσότητες που σχετίζονται με την επανάληψη k δηλώνονται με ένα άνω δείκτη σε παρενθέσεις. Πχ. $\mathbf{x}^{(k)}$ . Τα χρησιμοποιούμενα σύμβολα και η σημασία τους είναι:

**g** Το διάνυσμα των πρώτων παραγώγων της συνάρτησης με στοιχεία: $g_i = \frac{\partial f}{\partial x_i}$

**G** Ο πίνακας των δεύτερων παραγώγων της συνάρτησης με στοιχεία $G_{ij} = \frac{\partial^2 f}{\partial x_i \, \partial x_i}$

**H** Ο αντίστροφος του πίνακα των δευτέρων παραγώγων με στοιχεία $H_{ij} = \left[\mathbf{G}^{-1}\right]_{ij}$

## 10.2 Τερματισμός των ελαχιστοποιητών Εντολές TARGET και NOTARGET

Όλοι οι αλγόριθμοι ελαχιστοποίησης τερματίζουν τη λειτουργία τους όταν ξεπεράσουν ένα δοσμένο αριθμό κλήσεων της αντικειμενικής συνάρτησης. Σε αυτή την περίπτωση ο χρήστης ενημερώνεται από το μήνυμα:

```
...  ALL CALLS EXAUSTED  ...
```

Οι μέθοδοι BFGS, DFP, CONGRA, ROLL και RANDOM τερματίζουν τη λειτουργία τους, όταν φτάσουν σε μιά τιμή-στόχο. Ο έλεγχος αυτής της τιμής γίνεται με τις εντολές:

| | |
|---|---|
| TARGET | Θέτει την τιμή-στόχο γιά τις μεθόδους BFGS, DFP, CONGRA, ROLL και RANDOM. |
| NOTARGET | Ακυρώνει την τιμή-στόχο. |

Ο χρήστης ενημερώνεται για την επίτευξη της τιμής-στόχου με το διαγνωστικό:

```
...  TARGET VALUE HAS BEEN REACHED  ...
```

Επιπλέον τερματισμός των ανωτέρω μεθόδων επέρχεται όταν δεν υπάρχει επαρκής μείωση της τιμής της συνάρτησης σε κάποια επανάληψη. Η επαρκής μείωση ορίζεται από:

$$\left| f^{(k+1)} - f^{(k)} \right| \geq \varepsilon \, L \max\left( \left| f^{(k+1)} \right| , \left| f^{(k)} \right| \right)$$

όπου $f^{(k)}$ και $f^{(k+1)}$ είναι οι τιμές της συνάρτησης στο τέλος δύο διαδοχικών επαναλήψεων, L ο αριθμός των δαπανηθέντων κλήσεων στην επανάληψη k+1 και ε ένας προεπιλεγμένος πραγματικός αριθμός στο διάστημα [0,1]. Η τιμή ε=0 (που είναι η προκαθορισμένη τιμή) καταργεί τον έλεγχο αυτό. Ο χρήστης ενημερώνεται για αυτό τον τρόπο τερματισμού από το μήνυμα:

```
...  CONVERGENCE SLOWER THAN SPECIFIED  ...
```

Ορισμένοι από τους ελαχιστοποιητές χρησιμοποιούν και επιπλέον κριτήρια για τερματισμό. Αυτά αναφέρονται όπου χρειάζεται.

## 10.3 Κοινές παράμετροι

Στους πίνακες ελέγχου όλων των ελαχιστοποιητών υπάρχουν οι παρακάτω παράμετροι:

`NUMBER OF CALLS` Προσεγγιστικός μέγιστος επιτρεπτός αριθμός κλήσεων της αντικειμενικής συνάρτησης. Στον αριθμό αυτό συμπεριλαμβάνονται και οι κλήσεις που θα δαπανηθούν για τον υπολογισμό των παραγώγων όταν αυτές πρέπει να προσεγγιστούν αριθμητικά.

`PRINTOUT SELECTION` Επιλογή της εξόδου από τη μέθοδο. Επιτρεπτές τιμές είναι:

- 0 Η μέθοδος δεν παράγει καθόλου έξοδο.
- 1 Η μέθοδος αναφέρει την τιμή της αντικειμενικής συνάρτησης κάθε φορά που βρίσκει ένα χαμηλότερο σημείο όπως φαίνεται στον πίνακα 12. Αυτή είναι η προκαθορισμένη συμπεριφορά.
- 2 Η μέθοδος αναφέρει την τιμή της αντικειμενικής συνάρτησης και τις τιμές όλων των παραμέτρων κάθε φορά που βρίσκει χαμηλότερο σημείο.

`CANCEL-BUTTON` Ελέγχει την εκτέλεση της μεθόδου μετά από τις αλλαγές των παραμέτρων στον πίνακα ελέγχου. Επιτρεπτές τιμές είναι:

- 0 Οι αλλαγές του χρήστη στις παραμέτρους του πίνακα ελέγχου καταχωρούνται, αλλά η μέθοδος ελαχιστοποίησης δεν εκτελείται.
- 1 Η μέθοδος ελαχιστοποίησης εκτελείται. Αυτή είναι η προκαθορισμένη συμπεριφορά.

Οι μέθοδοι ελαχιστοποίησης που χρησιμοποιούν πρώτες παραγώγους (`BFGS`, `DFP`, `CONGRA` και `TOLMIN`) περιέχουν τις παραμέτρους ελέγχου:

`ERROR-BOUND` Επιτρεπτό ποσοστό σφάλματος κατά την αριθμητική προσέγγιση των παραγώγων. Επιτρεπτές τιμές είναι μεταξύ 0 και 1.

`GRADIENT-MODE` Ο τρόπος υπολογισμού των παραγώγων. Επιτρεπτές τιμές είναι:

- 0 Κατά την εκτέλεση της μεθόδου, οι παράγωγοι υπολογίζονται διαρκώς με τον τρόπο που έχει προκαθορίσει ο χρήστης (`FAST`,

QUAD, NUMER, ANAL ή MIXED). Αυτή είναι η προκαθορισμένη συμπεριφορά.

1 Καθώς το τοπικό ελάχιστο προσεγγίζεται απαιτούνται παράγωγοι μεγαλύτερης ακρίβειας. Χρησιμοποιείται η αυτόματη αλλαγή παραγώγων όπως περιγράφεται στην παράγραφο 5.3.

Οι μέθοδοι ελαχιστοποίησης που χρησιμοποιούν γραμμικές αναζητήσεις περιέχουν την παράμετρο ελέγχου:

TOLERANCE Η ακρίβεια με την οποία γίνεται η γραμμική αναζήτηση. Επιτρεπτές τιμές είναι μεταξύ 0 και 1.

## 10.4 BFGS

0) Δίνεται αρχική τιμή στον πίνακα **H** και υπολογίζεται το διάνυσμα των πρώτων παραγώγων $\mathbf{g}^{(1)}$.

Κατόπιν για την επανάληψη k:

1) Υπολογισμός της διεύθυνσης $\mathbf{s}^{(k)} = -\mathbf{H}^{(k)}\,\mathbf{g}^{(k)}$
2) Γραμμική αναζήτηση στη διεύθυνση $\mathbf{s}^{(k)}$ που δίνει $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}$
3) Ενημέρωση του $\mathbf{H}^{(k)}$ από τον τύπο (οι άνω δείκτες (k) έχουν παραλειφθεί χάριν απλότητος):

$$\mathbf{H}^{(k+1)} = \mathbf{H} + \left(1 + \frac{\boldsymbol{\gamma}^T \mathbf{H}\, \boldsymbol{\gamma}}{\boldsymbol{\delta}^T \boldsymbol{\gamma}}\right)\frac{\boldsymbol{\delta}\,\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T \boldsymbol{\gamma}} - \frac{\boldsymbol{\delta}\,\boldsymbol{\gamma}^T \mathbf{H} + \mathbf{H}\,\boldsymbol{\gamma}\,\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T \boldsymbol{\gamma}}$$

με $\boldsymbol{\gamma}^{(k)} = \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}$

Ο πίνακας ελέγχου της BFGS φαίνεται στον πίνακα 12. Οι παράμετροι που περιέχει είναι:

NUMBER OF CALLS Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3).

TOLERANCE Ακρίβεια της γραμμικής αναζήτησης (βλέπε παράγραφο 10.3).

ERROR-BOUND Επιτρεπτό σφάλμα κατά τον υπολογισμό των παραγώγων (βλέπε παράγραφο 10.3).

USE/RECALCULATE GRADIENT Ελέγχει τον υπολογισμό ή όχι των πρώτων παραγώγων πρίν την έναρξη των επαναλήψεων, στο βήμα 0 του αλγορίθμου. Επιτρεπτές τιμές είναι:

0 Οι παράγωγοι $\mathbf{g}^{(1)}$ υπολογίζονται πρίν την έναρξη των επαναλήψεων.

1 Οι παράγωγοι δεν υπολογίζονται πρίν την έναρξη των επαναλήψεων, αλλά χρησιμοποιείται η τευλευταία υπολογεισθείσα τιμή.

USE/RESET HESSIAN — Ελέγχει την αρχική τιμή που αποδίδεται στον πίνακα **H**, πρίν αρχίσουν οι επαναλήψεις, στο βήμα 0 του αλγορίθμου. Επιτρεπτές τιμές είναι:

0 Ο μοναδιαίος πίνακας δίνεται σαν αρχική τιμή του πίνακα **H**.

1 Ο πίνακας **H** διατηρεί τις τιμές που είχε μετά το τέλος της προηγούμενης ελαχιστοποίησης με BFGS ή DFP.

PRINTOUT SELECTION — Έλεγχος εξόδου (βλέπε παράγραφο 10.3).

WALL-PARAMETER — Όταν σε μία από τις παραμέτρους $x_i$ της συνάρτησης έχουν τεθεί όρια (με χρήση της εντολής MARGIN), η απόσταση της $x_i$ από το όριο είναι μικρότερη από $\left|\frac{x_i}{10^w}\right|$ και η μερική παράγωγος $g_i$ δείχνει ότι η $x_i$ τείνει να πλησιάσει περισσότερο στο όριο, η $x_i$ σταθεροποιείται προσωρινά έτσι ώστε να μείνει μέσα στά όρια. Οι τιμές του w πρέπει να είναι $w > 0$.

CANCEL BUTTON — Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3).

GRADIENT MODE — Τρόπος υπολογισμού παραγώγων (βλέπε παράγραφο 10.3).

MIN. RELATIVE DROP — Ελάχιστη επιτρεπτή μείωση της συνάρτησης σε μία επανάληψη (βλέπε παράγραφο 10.3).

## 10.5 DFP

0) Δίνεται αρχική τιμή στον πίνακα **H** και υπολογίζεται το διάνυσμα των πρώτων παραγώγων $\mathbf{g}^{(1)}$.

Κατόπιν για την επανάληψη k:

1) Υπολογισμός της διεύθυνσης $\mathbf{s}^{(k)} = -\mathbf{H}^{(k)}\,\mathbf{g}^{(k)}$
2) Γραμμική αναζήτηση στη διεύθυνση $\mathbf{s}^{(k)}$ που δίνει $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}$
3) Ενημέρωση του $\mathbf{H}^{(k)}$ από τον τύπο (οι άνω δείκτες (k) έχουν παραλειφθεί χάριν απλότητος):

$$\mathbf{H}^{(k+1)} = \mathbf{H} + \frac{\boldsymbol{\delta}\,\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\,\boldsymbol{\gamma}} - \frac{\mathbf{H}\,\boldsymbol{\gamma}\,\boldsymbol{\gamma}^T\,\mathbf{H}}{\boldsymbol{\gamma}^T\,\mathbf{H}\,\boldsymbol{\gamma}}$$

με $\boldsymbol{\gamma}^{(k)} = \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}$

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της DFP είναι:

| | |
|---|---|
| NUMBER OF CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| TOLERANCE | Ακρίβεια της γραμμικής αναζήτησης (βλέπε παράγραφο 10.3). |
| ERROR-BOUND | Επιτρεπτό σφάλμα κατά τον υπολογισμό των παραγώγων (βλέπε παράγραφο 10.3). |
| USE/RECALCULATE GRADIENT | Όπως γιά την BFGS (βλέπε παράγραφο 10.4). |
| USE/RESET HESSIAN | Όπως γιά την BFGS (βλέπε παράγραφο 10.4). |
| PRINTOUT SELECTION | Έλεγχος εξόδου (βλέπε παράγραφο 10.3). |
| WALL-PARAMETER | Όπως γιά την BFGS (βλέπε παράγραφο 10.4). |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |
| GRADIENT MODE | Τρόπος υπολογισμού παραγώγων (βλέπε παράγραφο 10.3). |
| MIN. RELATIVE DROP | Ελάχιστη επιτρεπτή μείωση της συνάρτησης σε μία επανάληψη (βλέπε παράγραφο 10.3). |

## 10.6 CONGRA

0) Υπολογίζεται το διάνυσμα των πρώτων παραγώγων $\mathbf{g}^{(1)}$ και τίθεται $\mathbf{s}^{(1)} = -\mathbf{g}^{(1)}$.

Γιά την επανάληψη k:

1) Υπολογισμός της διεύθυνσης $\mathbf{s}^{(k)} = -\mathbf{g}^{(k)} + \beta^{(k)}\,\mathbf{s}^{(k-1)}$
όπου εάν χρησιμοποιηθεί ο τύπος των Polak και Ribiere [8], είναι:

$$\beta^{(k)} = \frac{\left(\mathbf{g}^{(k)} - \mathbf{g}^{(k-1)}\right)^{T}\,\mathbf{g}^{(k)}}{\mathbf{g}^{(k)T}\,\mathbf{g}^{(k-1)}}$$

ενώ αν χρησιμοποιηθεί ο τύπος των Fletcher και Reeves [16], είναι:

$$\beta^{(k)} = \frac{\mathbf{g}^{(k)T}\,\mathbf{g}^{(k)}}{\mathbf{g}^{(k-1)T}\,\mathbf{g}^{(k-1)}}$$

2) Γραμμική αναζήτηση στη διεύθυνση $\mathbf{s}^{(k)}$ που θα δώσει το $\mathbf{x}^{(k+1)}$

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της CONGRA είναι:

| | |
|---|---|
| NUMBER OF CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| TOLERANCE | Ακρίβεια της γραμμικής αναζήτησης (βλέπε παράγραφο 10.3). |
| ERROR-BOUND | Επιτρεπτό σφάλμα κατά τον υπολογισμό των παραγώγων (βλέπε παράγραφο 10.3). |
| USE/RECALCULATE GRADIENT | Όπως γιά την BFGS (βλέπε παράγραφο 10.4). |

| | |
|---|---|
| PRINTOUT SELECTION | Έλεγχος εξόδου (βλέπε παράγραφο 10.3). |
| WALL-PARAMETER | Όπως γιά την BFGS (βλέπε παράγραφο 10.4). |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |
| GRADIENT MODE | Τρόπος υπολογισμού παραγώγων (βλέπε παράγραφο 10.3). |
| MIN. RELATIVE DROP | Ελάχιστη επιτρεπτή μείωση της συνάρτησης σε μία επανάληψη (βλέπε παράγραφο 10.3). |
| METHOD | Επιλογή του τύπου γιά το $\beta^{(k)}$. Επιτρεπτές τιμές είναι:<br>0 Χρησιμοποιείται ο τύπος των Polak και Ribiere. Αυτή είναι η προκαθορισμένη τιμή.<br>1 Χρησιμοποιείται ο τύπος των Fletcher και Reeves. |

## 10.7 ROLL

Έστω $\mathbf{x}^c$ το τρέχων σημείο και $f_c = f(\mathbf{x}^c)$ η τιμή της συνάρτησης. Έστω **s** ένα διάνυσμα δοκιμαστικών βημάτων σε κάθε διεύθυνση. Παρακάτω περιγράφεται ένας κύκλος λειτουργίας της ROLL που επαναλαμβάνεται γιά όλες τις παραμέτρους της συνάρτησης (i=1,N)

1) Επιλογή ενός δοκιμαστικού σημείου $\mathbf{x}^t$, έτσι ώστε $x_j^t = x_j^c$ γιά όλα τα $j \neq i$ και $x_i^t = x_i^c + s_i$
2) Υπολογισμός της $f_+ = f(\mathbf{x}^t)$
3) Εάν $f_+ < f_c$ θέτουμε $\mathbf{x}^c = \mathbf{x}^t$, $f_c = f_+$, $s_i = \alpha\, s_i$ και συνεχίζουμε από το βήμα 1 με το επόμενο i.
4) Εάν $f_+ \geq f_c$ επιλέγουμε ένα άλλο δοκιμαστικό σημείο $\mathbf{x}^t$, με $x_j^t = x_j^c$ γιά όλα τα $j \neq i$ και $x_i^t = x_i^c - s_i$
5) Υπολογισμός της $f_- = f(\mathbf{x}^t)$
6) Εάν $f_- < f_c$ θέτουμε $\mathbf{x}^c = \mathbf{x}^t$, $f_c = f_-$, $s_i = -\alpha\, s_i$ και συνεχίζουμε από το βήμα 1 με το επόμενο i.
7) Υπολογισμός ενός κατάλληλου βήματος ως: $s_i = \frac{1}{2} \frac{f_+ - f_-}{f_+ + f_- - 2f_c} s_i$

Άν μετά ένα πλήρη κύκλο δεν παρατηρηθεί πρόοδος, πραγματοποιείται μιά γραμμική αναζήτηση στη διεύθυνση **s**.

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της ROLL είναι:

| | |
|---|---|
| NUMBER OF CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| TOLERANCE | Ακρίβεια της γραμμικής αναζήτησης (βλέπε παράγραφο 10.3). |
| STEP FACTOR | Η παράμετρος α που ρυθμίζει το μέγεθος των βημάτων στα σημεία 3 και 6 του αλγορίθμου. |

FAILURES ALLOWED — Ο επιτρεπτός αριθμός διαδοχικών αποτυχημένων κύκλων της μεθόδου. Η μέθοδος τερματίζει τη λειτουργία της όταν υπάρξουν τόσες διαδοχικές αποτυχίες.

PRINTOUT SELECTION — Έλεγχος εξόδου (βλέπε παράγραφο 10.3).

WALL-PARAMETER — Όπως γιά την BFGS (βλέπε παράγραφο 10.4).

CANCEL BUTTON — Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3).

MIN. RELATIVE DROP — Ελάχιστη επιτρεπτή μείωση της συνάρτησης σε ένα κύκλο (βλέπε παράγραφο 10.3).

## 10.8 RANDOM

Έστω $\mathbf{x}^c$ το τρέχων σημείο και $f_c = f(\mathbf{x}^c)$ η τιμή της συνάρτησης. Έστω **s** ένα διάνυσμα δοκιμαστικών βημάτων σε κάθε διεύθυνση με $s_i > 0$.

1) Κατασκευάζουμε ένα κουτί N-διαστάσεων γύρω από το σημείο $\mathbf{x}^c$ με άκρα $x_i^c - s_i$ και $x_i^c + s_i$ σε κάθε διευθυνση.
2) Ένα δοκιμαστικό σημείο $\mathbf{x}^t$ δειγματίζεται τυχαία μέσα στο κουτί.
3) Υπολογίζεται η τιμή της συνάρτησης $f_t = f(\mathbf{x}^t)$
4) Εάν $f_t < f_c$ θέτουμε $\mathbf{x}^c = \mathbf{x}^t$, $f_c = f_t$ και δειγματίζουμε νέο σημείο.
5) Εάν $f_t \geq f_c$ επαναπροσδιορίζουμε το κουτί με μία από τις παρακάτω μεθόδους:
   α) Αποκλεισμός όγκου. Αν το αρχικό κουτί οριζόταν από τα διαστήματα $( l_i , r_i )$, το νέο κουτί $( l_i' , r_i' )$ θα είναι:

$$( l_i' , r_i' ) = \begin{cases} ( l_i , x_i^t ) & \text{εάν } x_i^c < x_i^t \\ ( x_i^t , r_i ) & \text{εάν } x_i^c > x_i^t \\ ( l_i , r_i ) & \text{εάν } x_i^c = x_i^t \end{cases}$$

   β) Μείωση πλευρών. Οι πλευρές του κουτιού πολλαπλασιάζονται με μία προκαθορισμένη σταθερά, κάθε φορά που συμπληρώνεται ένας αριθμός αποτυχιών. Τίθεται δηλαδή $s_i = \alpha\, s_i$

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της RANDOM είναι:

NUMBER OF CALLS — Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3).

ACTIVATE/DEACTIVATE
VOLUME EXCLUSION — Ελέγχει τον επαναπροσδιορισμό του κουτιού. Επιτρεπτές τιμές είναι:

- 0 Χρησιμοποιείται ο αποκλεισμός όγκου όπως περιγράφεται στο βήμα 5α. Αυτή είναι η προκαθορισμένη συμπεριφορά.

| | |
|---|---|
| | 1 Χρησιμοποιείται η μείωση πλευρών όπως περιγράφεται στο βήμα 5β. |
| STEP FACTOR | Στην περίπτωση που έχει επιλεγεί η μείωση πλευρών, αυτός είναι ο αριθμός α που ελέγχει τη μείωση. |
| CYCLE-SIZE | Ο αριθμός των διαδοχικών δειγματισμών στο βήμα 2. Αυτός ο αριθμός των διαδοχικών δειγματισμών, ονομάζεται "κύκλος". |
| NUMBER OF CONSECUTIVE CYCLE FAILURES ALLOWED | Ο αριθμός των διαδοχικών "κύκλων αποτυχίας" πριν η μέθοδος τερματίσει τη λειτουργία της. |
| LINE-SEARCH PERIOD | Είναι δυνατόν να πραγματοποιηθεί μία γραμμική αναζήτηση στη διεύθυνση που παρατηρείται πρόοδος. Η γραμμική αναζήτηση πραγματοποιείται κάθε ένα ορισμένο αριθμό κύκλων, που ελέγχεται από αυτή την παράμετρο. Θέτοντας την παράμετρο αυτή σε 0, δεν πραγματοποιείται γραμμική αναζήτηση. |
| PRINTOUT SELECTION | Έλεγχος εξόδου (βλέπε παράγραφο 10.3). |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |
| TOLERANCE | Ακρίβεια της γραμμικής αναζήτησης (βλέπε παράγραφο 10.3). |
| MIN. RELATIVE DROP | Ελάχιστη επιτρεπτή μείωση της συνάρτησης σε μία επανάληψη (βλέπε παράγραφο 10.3). |

## 10.9 SIMPLEX

1) Ξεκινώντας από το τρέχον σημείο $\mathbf{x}^c$ κατασκευάζουμε τις κορυφές simplex $\mathbf{p}_i$, i=0,N (με $\mathbf{p}_0 = \mathbf{x}^c$ ) και τις αντίστοιχες τιμές της συνάρτησης $y_i = f(\mathbf{p}_i)$.
2) Εύρεση της χαμηλότερης και υψηλότερης τιμής $y_{low}$ και $y_{high}$ και των αντίστοιχων κορυφών $\mathbf{p}_{low}$ και $\mathbf{p}_{high}$ :

   $y_{low} = \min \{ y_i, i=0,N \}$ $\quad y_{high} = \max \{ y_i, i=0,N \}$

   Υπολογισμός του κεντροειδούς $\mathbf{p}^c$ όλων των $\mathbf{p}_i$, εκτός από το $\mathbf{p}_{high}$.
   Θέτουμε $\mathbf{p}^* = (1+\alpha)\,\mathbf{p}^c - \alpha\,\mathbf{p}_{high}$ και υπολογίζουμε το $y^* = f(\mathbf{p}^*)$
3) Εάν $y^* < y_{low}$ τότε θέτουμε $\mathbf{p}^{**} = (1+\gamma)\,\mathbf{p}^* - \gamma\,\mathbf{p}^c$ και υπολογίζουμε το $y^{**} = f(\mathbf{p}^{**})$
   Εάν $y^{**} < y^*$ τότε αντικαθιστούμε το $\mathbf{p}_{high}$ από το $\mathbf{p}^{**}$ ειδάλως αντικαθιστούμε το $\mathbf{p}_{high}$ από το $\mathbf{p}^*$.
   Η διαδικασία συνεχίζεται στο βήμα 5 με έλεγχο του κριτηρίου σύγκλισης.
4) Εάν $y^* > y_i$ γιά όλα τα i εκτός από το $y_{high}$ τότε
   4α) Εάν $y^* < y_{high}$ αντικαθιστούμε το $\mathbf{p}_{high}$ από το $\mathbf{p}^*$.
   Θέτουμε $\mathbf{p}^{**} = (1-\beta)\,\mathbf{p}^c - \beta\,\mathbf{p}_{high}$ και υπολογίζουμε το $y^{**} = f(\mathbf{p}^{**})$.
   4β) Εάν $y^{**} > y_{high}$ τότε αντικαθιστούμε τα $\mathbf{p}_i$ από το $\dfrac{\mathbf{p}_i + \mathbf{p}_{low}}{2}$ γιά i=0,N
   αλλιώς αντικαθιστούμε το $\mathbf{p}_{high}$ από το $\mathbf{p}^{**}$.

Ειδάλως αντικαθιστούμε το $\mathbf{p}_{high}$ από το $\mathbf{p}^*$.

5) Ελέγχουμε το κριτήριο σύγκλισης. Εάν ικανοποιείται επιστρέφουμε τα $y_{low}$ , $\mathbf{p}_{low}$ αλλιώς επαναλαμβάνουμε από το βήμα 2. Χρησιμοποιείται το κριτήριο σύγκλισης:

$$\frac{1}{N+1} \sum_{i=0}^{N} |y_i - f(\mathbf{p}^c)| < \varepsilon$$

όπου ε είναι ένας προκαθορισμένος αριθμός. Οι τιμές των σταθερών α, β και γ είναι α=1, β=$\frac{1}{2}$ και γ=2 όπως προτείνεται στην αναφορά [17].

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της SIMPLEX είναι:

| | |
|---|---|
| INITIALIZATION SCHEME | Καθορίζει τον τρόπο με τον οποίο θα υπολογιστούν οι αρχικές κορυφές simplex $\mathbf{p}_i$ στο βήμα 1. Επιτρεπτές τιμές είναι:<br>1 Γίνεται γραμμική αναζήτηση σε κάθε μία από τις διευθύνσεις και το ελάχιστο που εντοπίζεται τίθεται ως τιμή για τα $\mathbf{p}_i$ και $y_i$ αντίστοιχα. Αυτή είναι η προκαθορισμένη τιμή.<br>2 Όταν έχουν τεθεί όρια σε όλες τις παραμέτρους $x_i$, είναι δυνατόν να αποδοθούν τυχαίες αρχικές τιμές στα $\mathbf{p}_i$. |
| INITIALIZATION TOLERANCE | Εάν επιλεγεί η γραμμική αναζήτηση για την απόδωση αρχικών τιμών, αυτή ειναι η απαιτούμενη ακρίβεια. |
| INIT. CALLS/VARIABLE | Εάν επιλεγεί η γραμμική αναζήτηση για την απόδωση αρχικών τιμών, αυτός είναι ο μέγιστος αριθμός κλήσεων που μπορεί να γίνει σε κάθε διεύθυνση. |
| SIMPLEX TOLERANCE | Καθορίζει την παράμετρο ε στο κριτήριο σύγκλισης. |
| SIMPLEX CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| PRINTOUT SELECTION | Έλεγχος εξόδου (βλέπε παράγραφο 10.3). |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |

## 10.10 TOLMIN

Περιγράφεται αναλυτικά στις αναφορές [19,20].

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της TOLMIN είναι:

| | |
|---|---|
| NUMBER OF CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| TERMINATION ACCURACY | Η απαιτούμενη ακρίβεια για τον τερματισμό της μεθόδου [20]. |
| ERROR-BOUND | Επιτρεπτό σφάλμα κατά τον υπολογισμό των παραγώγων (βλέπε παράγραφο 10.3). |
| PRINTOUT SELECTION | Έλεγχος εξόδου (βλέπε παράγραφο 10.3). |

| | |
|---|---|
| GRADIENT MODE | Τρόπος υπολογισμού παραγώγων (βλέπε παράγραφο 10.3). |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |

## 10.11 LEVE

Περιγράφεται αναλυτικά στις αναφορές [21,22]. Γιά να χρησιμοποιηθεί πρέπει να έχουμε θέσει το Merlin στον τρόπο λειτουργίας SOS (βλέπε παράγραφο 6.2) και να έχουμε προγραμματίσει την SUBROUTINE SUBSUM (παράγραφος 3.3).

Οι παράμετροι που περιέχονται στον πίνακα ελέγχου της LEVE είναι:

| | |
|---|---|
| NUMBER OF CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| F-TOLERANCE | Κριτήριο τερματισμού βασισμένο στις τιμές της συνάρτησης [22]. |
| X-TOLERANCE | Κριτήριο τερματισμού βασισμένο στις τιμές των παραμέτρων [22]. |
| G-TOLERANCE | Κριτήριο τερματισμού βασισμένο στις τιμές των παραγώγων [22]. |
| STEP FACTOR | Συντελεστής αλλαγής βήματος [22]. |
| FUNCTION ACCURACY | Η ακρίβεια με την οποία υπολογίζεται η συνάρτηση του χρήστη (SUBROUTINE SUBSUM). |
| PRINTOUT SELECTION | Έλεγχος εξόδου (βλέπε παράγραφο 10.3). |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |

## 10.12 AUTO

Εκτός από τους ήδη αναφερθέντες αλγόριθμους ελαχιστοποίησης, το Merlin περιέχει την εντολή AUTO που είναι ένας συνδιασμός των ROLL, SIMPLEX, BFGS και RANDOM. Η AUTO δοκιμάζει διαδοχικά τους τέσσερεις ελαχιστοποιητές και χρησιμοποιεί κάθε φορά αυτόν με την καλύτερη απόδοση.

Ο πίνακας ελέγχου της AUTO περιέχει τις παραμέτρους:

| | |
|---|---|
| NUMBER OF CALLS | Προσεγγιστικός μέγιστος αριθμός κλήσεων (βλέπε παράγραφο 10.3). |
| TARGET VALUE | Τιμή στόχος. Όταν η τιμή της συνάρτησης γίνει μικρότερη από την τιμή στόχο η διαδικασία AUTO θα τερματιστεί. |
| CANCEL BUTTON | Ακύρωση εκτέλεσης (βλέπε παράγραφο 10.3). |
| GRADIENT MODE | Τρόπος υπολογισμού παραγώγων (βλέπε παράγραφο 10.3). |

# Κεφάλαιο 11

# Η γλώσσα προγραμματισμού MCL

## 11.1 Τι είναι η γλώσσα MCL

Όπως έχει ήδη αναφερθεί ο κύριος τρόπος χρήσης του Merlin είναι διαλογικός. Ο χρήστης εισάγει τις εντολές του από κάποιο τερματικό και παρατηρεί το αποτέλεσμα στην οθόνη. Γιά παράδειγμα, ο χρήστης μπορεί να διαπιστώσει ότι κάποιος ελαχιστοποιητής υπερτερεί έναντι των άλλων και να αποφασίσει να τον χρησιμοποιεί συνεχώς, ή (αφού οι αλγόριθμοι ελαχιστοποίησης ελέγχονται από παραμέτρους) να βρεί τις τιμές των παραμέτρων ελέγχου που δίνουν τα καλύτερα αποτελέσματα. Η διαδικασία αυτή είναι χρονοβόρα διότι συνήθως απαιτείται αρκετός πειραματισμός, κάτι που δεν είναι επιθυμητό.

Η γλώσσα προγραμματισμού MCL (Merlin Control Language) έχει φτιαχτεί κυρίως για να αντιμετωπίσει αυτό το πρόβλημα. Είναι μία κατανοητή καί εύκολη στη μάθηση γλώσσα που ελέγχει το Merlin δυναμικά, όπως και ο χρήστης από το τερματικό του, όμως δεν απαιτεί την ανάμιξη ή την παρέμβαση του χρήστη. Η MCL υποστηρίζει αριθμητικές πράξεις και συναρτήσεις (όπως η Fortran), περιέχει εντολές ελέγχου και επανάληψης, πολυδιάστατους πίνακες, εντολές εισόδου εξόδου, κα. Η δομή της γλώσσας περιγράφεται πλήρως στο [2].

## 11.2 Ο μεταγλωτιστής MCL

Για είναι σε θέση το Merlin να εκτελέσει ένα πρόγραμμα MCL, αυτό πρέπει να μεταγλωτιστεί σε μιά σειρά εντολών που το Merlin αναγνωρίζει και ονομάζονται "αντικειμενικός κώδικας Merlin" (Merlin Object Code). Τη λειτουργία αυτή αναλαμβάνει ο μεταγλωτιστής MCL. Ο μεταγλωτιστής MCL, αναμένει γιά είσοδο μιά σειρά από παραμέτρους στη μορφή:

```
παράμετρος=τιμή, παράμετρος=τιμή ...
```

Τυπικό παράδειγμα χρήσης του μεταγλωτιστή φαίνεται στον πίνακα 34.

Οι παράμετροι που δέχεται ο μεταγλωτιστής είναι:

`I` Το όνομα του αρχείου με το πρόγραμμα MCL που πρόκειται να μεταγλωτιστεί.

`B` Το όνομα του αρχείου που θα δεχτεί τον αντικειμενικό κώδικα Merlin. Η προκαθορισμένη τιμή είναι `MOC`.

`E` Το όνομα του αρχείου που θα δεχτεί τον κατάλογο με τα λάθη που πιθανώς θα βρεί ο μεταγλωτιστής στο πρόγραμμα MCL. Εάν δεν εισάγουμε τιμή για την παράμετρο `E`, τα λάθη θα εμφανιστούν στην οθόνη.

DEBUG Καθορίζει αν στον αντικειμενικό κώδικα Merlin θα συμπεριληφθούν πληροφορίες εκσφαλμάτωσης. Επί του παρόντος η μόνη πληροφορία εκσφαλμάτωσης που μπορεί να συμπεριληφθεί στον αντικειμενικό κώδικα Merlin (και που εμφανίζεται αν προκύψει κάποιο λάθος κατά την εκτέλεση του από το Merlin) είναι οι αριθμοί γραμμών του πηγαίου κώδικα MCL. Οι δύο επιτρεπτές τιμές για την παράμετρο αυτή είναι:

- TRUE Συμπεριλαμβάνονται οδηγίες εκσφαλμάτωσης στον αντικειμενικό κώδικα Merlin.
- FALSE Δεν συμπεριλαμβάνονται οδηγίες εκσφαλμάτωσης. Αυτή είναι η προκαθορισμένη τιμή.

BOUNDS Καθορίζει αν στον παραγόμενο αντικειμενικό κώδικα Merlin θα συμπεριληφθούν επιπλέον εντολές που θα ελέγχουν την ώρα της εκτέλεσης, αν οι δείκτες των πινάκων είναι μέσα στα δηλωμένα όρια. Επιτρεπτές τιμές για αυτή την παράμετρο είναι:

- TRUE Συμπεριλαμβάνεται κώδικας γιά τον έλεγχο των δεικτών των πινάκων.
- FALSE Δεν συμπεριλαμβάνεται κώδικας γιά τον έλεγχο των δεικτών των πινάκων. Αυτή είναι η προκαθορισμένη τιμή.

Ο μεταγλωτιστής MCL είναι γραμμένος σε ANSI-Fortran-77. Περίληψη δίνεται στον πίνακα 1. Πρόγραμμα MCL και τμήμα του παραγόμενου αντικειμενικού κώδικα Merlin φαίνεται στον πίνακα 35. Ο αντικειμενικός κώδικας Merlin και η λειτουργία του μεταγλωτιστή MCL περιγράφεται στο εγχειρίδιο [23] που συνοδεύει την [2].

Πίνακας 34: Τυπικό παράδειγμα χρήσης του μεταγλωτιστή MCL.

```
-------------------------------------------------------------------------------
                              M C L - 1.0  PLUS
-------------------------------------------------------------------------------
  ENTER COMPILATION PARAMETERS   - I, B, E, DEBUG, BOUNDS -
i=test  <cr>
STOP: MCLCOM COMPLETE.
```

Πίνακας 35: Πρόγραμμα MCL και τμήμα του παραγόμενου αντικειμενικού κώδικα Merlin.

```
var i; a                              $LENGTH
                                        7
a = 2                                 $PUSHC
                                        2
loop i from 1 to dim                  $PUSHC
    point (x.i=a)                         2.00000
end loop                              $POPCONT
                                      $PUSHC
shortdis                                  1.00000
```

## 11.3 Εκτέλεση προγραμμάτων MCL Η εντολή RUNMCL

Προγράμματα MCL που έχουν μεταφραστεί σε αντικειμενικό κώδικα Merlin, εκτελούνται με χρήση της εντολής RUNMCL. Δείγμα χρήσης της RUNMCL και αποτελέσματα από πρόγραμμα MCL φαίνονται στον πίνακα 36.

Πίνακας 36: Αποτελέσματα από την εκτέλεση του προγράμματος MCL του πίνακα 35, με χρήση της εντολής RUNMCL.

```
     //////     MERLIN   IS AT YOUR COMMAND !!!
runmcl  <cr>
 RUNMCL
 ENTER FILE-NAME
MOC  <cr>
 POINT
 POINT
 SHORTDIS
 ------------------------------------------------------------------------------
 NUMBER OF EVALUATIONS:    FUNCTION        GRADIENT
 ------------------------------------------------------------------------------
 TOTAL:                          3               0
 SINCE LAST RESET:               3               0
 ------------------------------------------------------------------------------
   1)           .......   2.0000000000000
   2)           .......   2.0000000000000
         VALUE.......     401.00000000000


     //////     MERLIN   IS AT YOUR COMMAND !!!
```

## 11.4 Οι εντολές LEVE και TOLMIN

Οι νέοι ελαχιστοποιητές TOLMIN και LEVE που παρουσιάζονται σε αυτή την έκδοση του Merlin υποστηρίζονται από την MCL. Έχουν υλοποιηθεί ως πολυπαραμετρικές εντολές MCL [2,23].

Η αντιστοιχία των προσδιοριστών MCL με τον πίνακα ελέγχου της TOLMIN είναι:

| Παράμετρος ελέγχου | Προσδιοριστής MCL |
|---|---|
| NUMBER OF CALLS | NOC |
| TERMINATION ACCURACY | ACC |
| ERROR-BOUND | ERR |
| PRINTOUT SELECTION | PRINT |
| GRADIENT MODE | MAD |
| CANCEL BUTTON | CANCEL |

Η αντιστοιχία των προσδιοριστών MCL με τον πίνακα ελέγχου της LEVE είναι:

| Παράμετρος ελέγχου | Προσδιοριστής MCL |
|---|---|
| NUMBER OF CALLS | NOC |
| F-TOLERANCE | FTOL |
| X-TOLERANCE | XTOL |
| G-TOLERANCE | GTOL |
| STEP FACTOR | FACTOR |
| FUNCTION ACCURACY | FACC |
| PRINTOUT SELECTION | PRINT |
| CANCEL BUTTON | CANCEL |

# Κεφάλαιο 12

# Διαμόρφωση του συστήματος. Το αρχείο CONFIG

Το Merlin είναι γραμμένο σε ANSI-Fortran-77 και δεν περιέχει ενσωματωμένα χαρακτηριστικά που να εξαρτώνται από κάποιο συγκεκριμένο υπολογιστή. Όμως την πρώτη φορά που το Merlin εγκαθίσταται σε ένα σύστημα ορισμένες παράμετροι πρέπει να καθοριστούν ορθά. (βλέπε επίσης την παράγραφο 2.1). Τέτοιες παράμετροι είναι οι αριθμοί των μονάδων εισόδου/εξόδου, ο μεγαλύτερος πραγματικός αριθμός που μπορεί να διαπραγματευτεί το σύστημα, κα. Ο μηχανισμός αλλαγής αυτών των παραμέτρων έχει ως εξής:

Όταν το Merlin ξεκινά, ελέγχει γιά την ύπαρξη ενός αρχείου με το όνομα `CONFIG`. Εάν το αρχείο βρεθεί, τότε αυτό ανοίγεται και διαβάζεται γραμμή προς γραμμή. Κάθε γραμμή του `CONFIG` περιέχει μία οδηγία διαμόρφωσης που αλλάζει την προκαθορισμένη συμπεριφορά του Merlin. Οι οδηγίες διαμόρφωσης, η δράση τους και οι αντίστοιχες προκαθορισμένες τιμές περιγράφονται παρακάτω. Στην περιγραφή που ακολουθεί το <x> αναφέρεται σε ένα πραγματικό αριθμό, τα <n> και <m> σε ακέραιο, ενώ το <filename> σε όνομα αρχείου.

## 12.1 BIGGER

| | |
|---|---|
| Σύνταξη: | `BIGGER <x>` |
| Σκοπός: | Καθορίζει (κατά απόλυτη τιμή) τον μέγιστο πραγματικό αριθμό που μπορεί να χειριστεί το σύστημα. |
| Προκαθορισμένη τιμή: | 1.E+38 |
| Σχόλια: | Η προκαθορισμένη τιμή είναι κατάλληλη για αριθμούς απλής ακρίβειας για τους περισσότερους μεταγλωτιστές. |

## 12.2 SMALLER

| | |
|---|---|
| Σύνταξη: | `SMALLER <x>` |
| Σκοπός: | Καθορίζει (κατά απόλυτη τιμή) τον μικρότερο πραγματικό αριθμό που μπορεί να χειριστεί το σύστημα χωρίς να προκύψει κατάσταση underflow. |
| Προκαθορισμένη τιμή: | 1.E-36 |
| Σχόλια: | Η προκαθορισμένη τιμή είναι κατάλληλη για αριθμούς απλής ακρίβειας για τους περισσότερους μεταγλωτιστές. |

## 12.3 LIMIT

Σύνταξη: `LIMIT <x>`

Σκοπός: Καθορίζει (κατά απόλυτη τιμή) τον πραγματικό αριθμό που θα χρησιμοποιηθεί σαν προκαθορισμένη τιμή για τα όρια των παραμέτρων της αντικειμενικής συνάρτησης, στις οποίες ο χρήστης δεν έχει θέσει όρια.

Προκαθορισμένη τιμή: 1.E+30

Σχόλια: Η προκαθορισμένη τιμή είναι κατάλληλη για αριθμούς απλής ακρίβειας για τους περισσότερους μεταγλωτιστές. Η τιμή αυτή δεν είναι κρίσιμη και γενικά πρέπει να είναι μερικές τάξεις μεγέθους μικρότερη από τον αριθμό `BIGGER`.

## 12.4 INPUT_UNIT

Σύνταξη: `INPUT_UNIT <n>`

Σκοπός: Καθορίζει τον αριθμό που θα χρησιμοποιηθεί σαν αριθμός μονάδος εισόδου στις εντολές εισόδου Fortran.

Προκαθορισμένη τιμή: 5

Σχόλια: Ο αριθμός αυτός πρέπει να τεθεί έτσι ώστε εντολές `read` από την αντίστοιχη μονάδα <n> να αναφέρονται στο πληκτρολόγιο. Ο αριθμός αυτός δεν χρησιμοποιείται όταν καθοριστεί αρχείο εισόδου με την οδηγία διαμόρφωσης `INPUT_FILE`.

## 12.5 OUTPUT_UNIT

Σύνταξη: `OUTPUT_UNIT <n>`

Σκοπός: Καθορίζει τον αριθμό που θα χρησιμοποιηθεί σαν αριθμός μονάδος εξόδου στις εντολές εξόδου Fortran.

Προκαθορισμένη τιμή: 6

Σχόλια: Ο αριθμός αυτός πρέπει να τεθεί έτσι ώστε εντολές `write` στην αντίστοιχη μονάδα <n> να αναφέρονται στην οθόνη. Ο αριθμός αυτός δεν χρησιμοποιείται όταν καθοριστεί αρχείο εξόδου με την οδηγία διαμόρφωσης `OUTPUT_FILE`.

## 12.6 INPUT_FILE

Σύνταξη: `INPUT_FILE <filename>`

Σκοπός: Καθορίζει το όνομα του αρχείου από όπου το Merlin θα διαβάσει την είσοδο.

Προκαθορισμένη τιμή: Καμία. Η είσοδος διαβάζεται από την μονάδα που καθορίστηκε με την οδηγία διαμόρφωσης `INPUT_UNIT`. Η μονάδα αυτή πρέπει να αντιστοιχεί στο πληκτρολόγιο.

## 12.7 OUTPUT_FILE

Σύνταξη: `OUTPUT_FILE <filename>`

Σκοπός: Καθορίζει το όνομα του αρχείου όπου το Merlin θα γράψει την έξοδο.

Προκαθορισμένη τιμή: Καμία. Η έξοδος γράφεται στην μονάδα που καθορίστηκε με την οδηγία διαμόρφωσης `OUTPUT_UNIT`. Η μονάδα αυτή πρέπει να αντιστοιχεί στην οθόνη.

## 12.8 PRECONNECT

Σύνταξη: `PRECONNECT`

Σκοπός: Καθορίζει ότι η είσοδος/έξοδος είναι προσυνδεμένη από το λειτουργικό σύστημα στις αντίστοιχες μονάδες εισόδου/εξόδου. Σαν αποτέλεσμα το Merlin δεν θα χρησιμοποιήσει εντολές `open` ή `close` γιά αυτά τα αρχεία. Αν όμως έχει καθοριστεί το όνομα κάποιου αρχείου για είσοδο ή έξοδο (με τις οδηγίες διαμόρφωσης `INPUT_FILE` ή `OUTPUT_FILE`), το οποίο δεν είναι ήδη ανοικτό κατά την κλήση του Merlin, τότε το Merlin θα προσδιορίσει ένα κατάλληλο αριθμό μονάδος (διαφορετικό από αυτό που δίνεται στις εντολές `INPUT_UNIT` ή `OUTPUT_UNIT`) και θα ανοίξει το αρχείο. Όταν το Merlin τελειώσει, θα κλείσει το αρχείο, αφήνοντάς το στην αρχική κατάσταση.

Προκαθορισμένη τιμή: Η είσοδος και έξοδος είναι προσυνδεμένες από το λειτουργικό σύστημα στις μονάδες εισόδου/εξόδου που καθορίζονται από τις οδηγίες διαμόρφωσης `INPUT_UNIT` και `OUTPUT_UNIT`.

## 12.9 NOPRECONNECT

Σύνταξη: NOPRECONNECT

Σκοπός: Καθορίζει ότι η είσοδος/έξοδος δεν είναι προσυνδεμένη από το λειτουργικό σύστημα στις αντίστοιχες μονάδες εισόδου/εξόδου. Σαν αποτέλεσμα το Merlin θα χρησιμοποιήσει εντολές `open` και `close` γιά αυτά τα αρχεία αφού προσδιορίσει έναν κατάλληλο αριθμό μονάδος (διαφορετικό από αυτό που δίνεται στις εντολές `INPUT_UNIT` ή `OUTPUT_UNIT`). Όταν το Merlin τελειώσει, θα κλείσει το αρχεία, αφήνοντάς τα στην αρχική κατάσταση.

Προκαθορισμένη τιμή: Η είσοδος και έξοδος είναι προσυνδεμένες από το λειτουργικό σύστημα στις μονάδες εισόδου/εξόδου που καθορίζονται από τις οδηγίες διαμόρφωσης `INPUT_UNIT` και `OUTPUT_UNIT`. Βλέπε επίσης και την οδηγία `PRECONNECT`.

## 12.10 UNIT_RANGE

Σύνταξη: `UNIT_RANGE <n> <m>`

Σκοπός: Καθορίζει την περιοχή των αριθμών μονάδων που επιτρέπεται να χρησιμοποιήσει το Merlin για εσωτερικές διαδικασίες εισόδου/εξόδου.

Προκαθορισμένη τιμή: 50 80

Σχόλια: Αν ένας αριθμός στην ανωτέρω περιοχή έχει ανατεθεί σε κάποιο αρχείο που είναι ήδη ανοικτό κατά την κλήση του υποπρογράμματος Merlin, ο αριθμός αυτός δεν θα χρησιμοποιηθεί.

## 12.11 NOHEADER

Σύνταξη: NOHEADER

Σκοπός: Οι αρχικές πληροφορίες που δίνονται κατά την εκκίνηση του Merlin (έκδοση, e-mail κλπ., βλέπε πίνακα 8) δεν εμφανίζονται.

Προκαθορισμένη τιμή: Οι πληροφορίες εμφανίζονται.

## 12.12 HEADER

Σύνταξη: HEADER

Σκοπός: Οι αρχικές πληροφορίες που δίνονται κατά την εκκίνηση του Merlin (έκδοση, e-mail κλπ., βλέπε πίνακα 8) εμφανίζονται.

Προκαθορισμένη τιμή: Οι πληροφορίες εμφανίζονται.

## 12.13 CSET

Σύνταξη: `CSET <set>`

Σκοπός: Καθορίζει το σύνολο χαρακτήρων που αναγνωρίζει το Merlin. Επιτρεπτές τιμές για το <set> είναι:

- `ASCII` Αναγνωρίζονται οι χαρακτήρες του συνόλου `ASCII`.
- `EBCDIC` Αναγνωρίζονται οι χαρακτήρες του συνόλου `EBCDIC`.
- `NORMAL` Αναγνωρίζονται μόνο κεφαλαία γράμματα. Οτιδήποτε εισάγεται με πεζούς χαρακτήρες δεν αναγνωρίζεται.

Προκαθορισμένη τιμή: `ASCII`

## 12.14 PRINTOUT

Σύνταξη: `PRINTOUT <print-option>`

Σκοπός: Καθορίζει το αρχικό επίπεδο εξόδου από το πρόγραμμα. Επιτρεπτές τιμές για το <print-option> είναι:

- `FULL` Παράγεται η μέγιστη δυνατή έξοδος με πλήθος διαγνωστικών μηνυμάτων και προτροπών.
- `HALF` Παράγεται μερική έξοδος με τα πιό σημαντικά διαγνωστικά μηνύματα και προτροπές.
- `NO` Δεν παράγεται καθόλου έξοδος.

Προκαθορισμένη τιμή: `FULL`

Σχόλια: Βλέπε επίσης και την παράγραφο 6.4.

## 12.15 PROLOG

Σύνταξη: `PROLOG <command>`

Σκοπός: Καθορίζει ότι η <command> (που υποτίθεται ότι είναι μία σωστή εντολή Merlin ή κλήση εντολοπακέτου), θα εκτελεστεί μόλις το Merlin ξεκινήσει.

Προκαθορισμένη τιμή: Καμία. Δεν εκτελείται πρόλογος.

## 12.16 EPILOG

Σύνταξη: `EPILOG <command>`

Σκοπός: Καθορίζει ότι η <command> (που υποτίθεται ότι είναι μία σωστή εντολή Merlin ή κλήση εντολοπακέτου), θα εκτελεστεί μόλις δοθεί στο Merlin μία από τις εντολές τερματισμού `STOP`, `RETURN` ή `QUIT` και πριν ο έλεγχος επιστρέψει στο καλών πρόγραμμα. Αν κάποια από αυτές τις εντολές δοθεί με μία οποιαδήποτε παράμετρο (π.χ. `STOP XX`), η <command> δεν θα εκτελεστεί.

Προκαθορισμένη τιμή: Καμία. Δεν εκτελείται επίλογος.

Σχόλια: Αν στη θέση της <command> δοθεί η κλήση ενός εντολοπακέτου (π.χ. `EPILOG .ENDIT`), είναι δυνατόν να δοθεί μία δεύτερη εντολή τερματισμού από το εντολοπακέτο. Σε αυτή την περίπτωση η <command> δεν εκτελείται ξανά, αλλά το Merlin τερματίζει τη λειτουργία του ή επιστρέφει στο καλών πρόγραμμα (ανάλογα με την αντίστοιχη εντολή τερματισμού) αμέσως.

## 12.17 MODE

Σύνταξη: `MODE <mode>`

Σκοπός: Καθορίζει ότι η λειτουργία του πακέτου θα γίνει με τον τρόπο <mode>. Επιτρεπτές τιμες είναι:

- `IAF` Διαλογικός τρόπος χρήσης. Οποιοδήποτε λάθος δεν τερματίζει το πρόγραμμα, αλλά επιστρέφει τον έλεγχο στον χρήστη γιά την εισαγωγή μιας νέας εντολής.
- `BATCH` Μη διαλογικός τρόπος χρήσης. Οποιοδήποτε λάθος τερματίζει τη λειτουργία του προγράμματος.

Προκαθορισμένη τιμή: `IAF`

Σχόλια: Εάν στο αρχείο `CONFIG` δεν υπάρχει η οδηγία διαμόρφωσης `MODE`, τότε κατά την εκκίνηση του Merlin, ο χρήστης θα ερωτηθεί διαλογικά για τον επιθυμητό τρόπο χρήσης.

## 12.18 FILE

Σύνταξη: `FILE <filename>`

Σκοπός: Τα περιεχόμενα του αρχείου `CONFIG`, από την επόμενη γραμμή και έως μιά γραμμή που θα περιέχει τους χαρακτήρες `...END` αντιγράφονται κατά την εκκίνηση του Merlin στο αρχείο <filename>.

Προκαθορισμένη τιμή: Καμία.

## 12.19 VAR

Σύνταξη: `VAR <n>`

Σκοπός: Καθορίζει τον αριθμό των παραμέτρων της αντικειμενικής συνάρτησης. Εάν χρησιμοποιηθεί η οδηγία διαμόρφωσης `VAR`, ακυρώνεται ο αριθμός των παραμέτρων που είχε δοθεί σαν είσοδος κατά την κλήση του υποπρογράμματος Merlin.

Προκαθορισμένη τιμή: Καμία. Εάν δεν υπάρχει οδηγία διαμόρφωσης `VAR`, χρησιμοποιείται η τιμή που δίνεται ώς είσοδος κατά την κλήση του υποπρογράμματος Merlin.

Σχόλια: Η οδηγία διαμόρφωσης `VAR` παρέχεται ώς ευκολία για αυτούς που αναπτύσουν τον κώδικα του Merlin. Δεν πρέπει να χρησιμοποιείται ως ο βασικός μηχανισμός καθορισμού του αριθμού των παραμέτρων της αντικειμενικής συνάρτησης.

## 12.20 TERMS

Σύνταξη: `TERMS <n>`

Σκοπός: Καθορίζει τον αριθμό των όρων της αντικειμενικής συνάρτησης, όταν αυτή έχει τη μορφή αθροίσματος τετραγώνων. Εάν χρησιμοποιηθεί η οδηγία διαμόρφωσης `TERMS`, ακυρώνεται ο αριθμός των όρων που είχε δοθεί σαν είσοδος κατά την κλήση του υποπρογράμματος Merlin.

Προκαθορισμένη τιμή: Καμία. Εάν δεν υπάρχει οδηγία διαμόρφωσης `TERMS`, χρησιμοποιείται η τιμή που δίνεται ώς είσοδος κατά την κλήση του υποπρογράμματος Merlin.

Σχόλια: Η οδηγία διαμόρφωσης `TERMS` παρέχεται ώς ευκολία για αυτούς που αναπτύσουν τον κώδικα του Merlin. Δεν πρέπει να χρησιμοποιείται ως ο βασικός μηχανισμός καθορισμού του αριθμού των όρων της αντικειμενικής συνάρτησης.

## 12.21 Σχόλια

Στο αρχείο `CONFIG` μπορούν να εισαχθούν επεξηγηματικά σχόλια. Αυτά εισάγονται με τη χρήση του `*` σαν πρώτου χαρακτήρα μιας γραμμής.

# Κεφάλαιο 13

# Επεκτάσεις

## 13.1 Μέγιστος αριθμός παραμέτρων και όρων

Ο μέγιστος αριθμός παραμέτρων της αντικειμενικής συνάρτησης καθορίζεται στον πηγαίο κώδικα Fortran με τη δήλωση:

```
PARAMETER ( MXV = 200 )
```

ενώ ο μέγιστος αριθμός όρων όταν η συνάρτηση είναι άθροισμα τετραγώνων καθορίζεται από τη δήλωση:

```
PARAMETER ( MXT = 200 )
```

Οι παραπάνω δηλώσεις εμφανίζονται σε διάφορα σημεία του πηγαίου κώδικα και χρησιμοποιούνται για τον ορισμό βοηθητικών μονοδιάστατων πινάκων.

# Βιβλιογραφία


1. G.A. Evangelakis, J.P. Rizos, I.E. Lagaris and I.N. Demetropoulos, Merlin - A Portable System for Multidimensional Minimization, Comp. Phys. Commun. 46 (1987) 401-415

2. C.S. Chassapis, D.G. Papageorgiou and I.E. Lagaris, MCL - Optimization Oriented Programming Language, Comp. Phys. Commun. 52 (1989) 223-239

3. D.G. Papageorgiou, C.S. Chassapis and I.E. Lagaris, Merlin-2.0 - Enhanced and Programmable Version, Comp. Phys. Commun. 52 (1989) 241-247

4. D.G. Papageorgiou and I.E. Lagaris, Merlin-2.1 - Double Precision, Comp. Phys. Commun. 58 (1990) 119-125

5. P.E. Gill, W. Murray and M.H. Wright, Practical Optimization, Academic Press, 1989

6. R. Fletcher, Practical Methodes of Optimization, Volume 1, Unconstrained Optimization, John Wiley, 1980

7. J.E Dennis jr. and R.B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice Hall, 1983

8. E. Polak, Computational Methods in Optimization: A Unified Approach, Academic Press, 1971

9. M.A. Wolfe, Numerical Methods for Unconstrained Optimization, Van Nostrand Reinhold Company Ltd., 1987

10. Programming Language FORTRAN, ANSI X3.9-1978, American National Standards Institute, New York

11. J.J More, B.S. Garbow and K.E. Hillstrom, Testing Unconstrainded Optimization Software, ACM Trans. Math. Softw. 7, (1981) 17-41

12. A Fortran-to-C Converter, S.I. Feldman, D.M. Gray, M.W. Maimone, N.L. Schryer, Computing Science Technical Report No. 149, AT&T Bell Laboratories

13. H.H Rosenbrock, An Automatic Method for Finding the Greatest or Least Value of a Function, Computer J. 3 (1960) 175-184

14. C.G. Broyden, The Convergence of a Class of Double Rank Minimization Algorithms, parts I and II, J. Inst. Maths. Applns., 6 (1970) 76-90 and 222-231
R. Fletcher, A New Approach to Variable Metric Algorithms, Computer J., 13 (1970) 317-322
D. Goldfarb, A Family of Variable Metric Methods Derived by Variational Means, Maths. Comput. 24 (1970) 23-26
D. Shanno, Conditioning of Quasi-Newton Methods for Function Minimization, Maths Comput., 24 (1970) 647-656

15. W.C. Davidon, Variable Metric Method for Minimization, AEC Res. and Dev. Report ANL-5990 (revised), 1959
R. Fletcher and M.J.D. Powell, A Rapidly Convergent Descent Method for Minimization, Computer J. 6 (1963) 163-168

16. R. Fletcher and C.M. Reeves, Function Minimization by Conjugate Gradients, Comp. J., 7 (1964) 149-154

17. J. Nedler and R. Mead, Computer J. 7 (1965) 308

18. K. Mika and Th. Chaves, Bericte der KFA, Julich No 1643 (1980)

19. M.J.D. Powell, A Tolerant Algorithm for Linearly Constrained Optimization Calculations, Mathematical Programming, 45, 1989,547-566

20. M.J.D. Powell, TOLMIN: A Fortran Package for Linearly Constrained Optimization Calculations, DAMTP 1989/NA2, Dept. of Applied Mathematics and Theoretical Physics, Univ. of Cambridge, June 1989.

21. M.D. Hebden, An Algorithm for Minimization Using Exact Second Derivatives, Atomic Energy Research Establishment report TP 515, Harwell, England.
J.J. More, the Levenberg-Marquardt algoritm: Implementation and Theory, Numerical Analysis, G.A. Watson ed., Lecture Notes in mathematics 630, Springer-Verlag.

22. J.J. More, B.S. Garbow, K.E. Hillstrom, User Guide for Minpack-1, ANL-80-74, August 1980

23. C.S. Chassapis, D.G. Papageorgiou and I.E. Lagaris, MCL and MCLCOM 1.0 Developers Manual, Comp. Phys. Commun. program ABHA (1989)

# Παράρτημα Α

# Οι εντολές Merlin

| Εντολή | Σελίδα | Εντολή | Σελίδα |
|---|---|---|---|
| ACCUM | 39 | ADJUST | 34 |
| ALIAS | 32 | ALIASDIS | 32 |
| ANAL | 25 | AUTO | 60 |
| BATCH | 30 | BFGS | 53 |
| CATALOG | 46 | CONGRA | 55 |
| CSET | 34 | DATASKIP | 44 |
| DATAWRITE | 44 | DEMARGIN | 36 |
| DFP | 54 | DISCARD | 45 |
| DISKOFF | 49 | DISKON | 49 |
| DISPOSKIP | 44 | DISPOWRITE | 44 |
| EVALUATE | 33 | FAST | 25 |
| FIX | 37 | FIXALL | 37 |
| FULLBACK | 43 | FULLPRINT | 31 |
| GENERAL | 30 | GERROR | 25 |
| GODFATHER | 33 | GRADCHECK | 25 |
| GRADDIS | 25 | GRAPH | 40 |
| HALFPRINT | 31 | HELP | 45 |
| HIDEOUT | 34 | IAF | 30 |
| INDEX | 33 | INIT | 41 |
| INSPECT | 44 | INTRODUCE | 45 |
| LASTBACK | 43 | LEVE | 60 |
| LIMITS | 29 | LIST | 45 |
| LOOSALL | 37 | LOOSE | 37 |
| MACRO | 48 | MAD | 27 |
| MARGIN | 36 | MEMO | 44 |
| MIXED | 25 | MODEDIS | 29 |
| NAME | 33 | NOBACK | 43 |
| NOEVAL | 33 | NOPRINT | 31 |
| NOTARGET | 51 | NUMER | 25 |
| PANELOFF | 31 | PANELON | 31 |
| PICK | 44 | POINT | 20 |
| QUAD | 25 | QUIT | 24 |
| RANDOM | 57 | RESET | 31 |
| RETURN | 24 | REVEAL | 34 |
| ROLL | 56 | RUNMCL | 63 |
| SHORTDIS | 21 | SIMPLEX | 58 |
| SOS | 30 | STEP | 34 |
| STEPALL | 34 | STEPDIS | 34 |
| STOP | 24 | TARGET | 51 |
| TOLMIN | 59 | UNALIAS | 32 |
| VALDIS | 21 | | |

# Παράρτημα Β

# Ευρετήριο πινάκων


| Πίνακας | Περιγραφή | Σελίδα |
|---|---|---|
| 1 | Περίληψη των Merlin και MCL | 7 |
| 2 | Υπολογιστικά συστήματα στα οποία εγκαταστάθηκε και λειτούργησε το πακέτο Merlin/MCL σε απλή και διπλή ακρίβεια | 12 |
| 3 | Υπόδειγμα της συνάρτησης funmin. Έχει κωδικοποιηθεί η συνάρτηση Rosenbrock | 13 |
| 4 | Υπόδειγμα του υποπρογράμματος granal. Έχουν κωδικοποιηθεί οι παράγωγοι της συνάρτησης Rosenbrock | 14 |
| 5 | Υπόδειγμα του υποπρογράμματος subsum. Έχουν κωδικοποιηθεί οι επιμέρους όροι $f_i$ της συνάρτησης Rosenbrock | 16 |
| 6 | Απαιτούμενος χώρος μνήμης για τις διάφορες μεθόδους ελαχιστοποίησης (N = αριθμός παραμέτρων, M = αριθμός όρων στο άθροισμα τετραγώνων ) | 16 |
| 7 | Υπόδειγμα κυρίως προγράμματος | 17 |
| 8 | Τα μηνύματα που παρουσιάζονται κατά την εκκίνηση του Merlin | 18 |
| 9 | Πλήρης διάλογος ανάθεσης αρχικών τιμών με την εντολή POINT | 20 |
| 10 | Τυπικό αποτέλεσμα από τη χρήση της εντολής SHORTDIS | 21 |
| 11 | Οι αλγόριθμοι ελαχιστοποίησης του Merlin | 22 |
| 12 | Τυπικό παράδειγμα χρήσης της μεθόδου BFGS | 23 |
| 13 | Ο πίνακας ελέγχου της BFGS, όπως θα παρουσιαζόταν κατά την δεύτερη κλήση της μεθόδου | 24 |
| 14 | Παράδειγμα χρήσης της εντολής GRADDIS | 26 |
| 15 | Δείγμα από τη χρήση της GRADCHECK | 26 |
| 16 | Αποτέλεσμα από χρήση των εντολών MODEDIS και LIMITS | 29 |
| 17 | Χρήση των εντολών ALIAS, ALIASDIS και UNALIAS | 32 |
| 18 | Χρήση των εντολών STEP και STEPDIS | 35 |
| 19 | Παράδειγμα χρήσης της εντολής MARGIN καθώς και εξέταση των ορίων αμέσως μετά με την εντολή SHORTDIS | 36 |
| 20 | Παράδειγμα χρήσης της εντολής DEMARGIN καθώς και εξέταση των ορίων αμέσως μετά με την εντολή SHORTDIS | 37 |
| 21 | Παράδειγμα χρήσης της εντολής FIX καθώς και εξέταση των παραμέτρων αμέσως μετά με την εντολή SHORTDIS | 38 |
| 22 | Παράδειγμα χρήσης της εντολής LOOSE καθώς και εξέταση των παραμέτρων αμέσως μετά με την εντολή SHORTDIS | 38 |

| | | |
|---|---|---|
| 23 | Δείγμα χρήσης της εντολής ACCUM | 39 |
| 24 | Τυπικό παράδειγμα χρήσης της GRAPH | 41 |
| 25 | Ανάθεση αρχικών τιμών με την εντολή INIT | 42 |
| 26 | Τρόπος αποθήκευσης μιάς εγγραφής στα αρχεία BACKUP, DATA, DISPO, STORE και INIPO | 43 |
| 27 | Χρήση της εντολής INSPECT. Το επιθεωρούμενο αρχείο BACKUP έχει 2 εγγραφές | 45 |
| 28 | Απόσπασμα του αρχείου HELP | 46 |
| 29 | Αποτέλεσμα από τη χρήση της εντολής CATALOG | 46 |
| 30 | Τυπικό παράδειγμα εντολοπακέτου δίσκου | 47 |
| 31 | Τυπικό παράδειγμα του αρχείου MACROF με δύο εντολοπακέτα | 48 |
| 32 | Κατασκευή εντολοπακέτου με την εντολή MACRO | 48 |
| 33 | Κλήση εντολοπακέτου | 49 |
| 34 | Τυπικό παράδειγμα χρήσης του μεταγλωτιστή MCL | 62 |
| 35 | Πρόγραμμα MCL και τμήμα του παραγόμενου αντικειμενικού κώδικα Merlin | 62 |
| 36 | Αποτελέσματα από την εκτέλεση του προγράμματος MCL του πίνακα 35, με χρήση της εντολής RUNMCL | 63 |